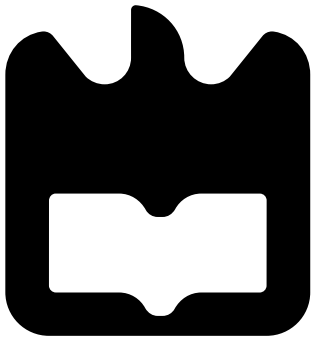
Universidade de Aveiro
Departamento de Electrónica, Telecomunicações e Informática,
2011

Gonçalo Alexandre Antunes Morais

Segurança em ambientes de proximidade
Security in proximity environments



Universidade de Aveiro
Departamento de Electrónica, Telecomunicações e Informática,
2011

Gonçalo Alexandre Antunes Morais

Segurança em ambientes de proximidade
Security in proximity environments

Thesis presented to University of Aveiro in order to achieve the necessary prerequisites to obtain the Master degree in Engineering of Computers and Telematics, under the scientific supervision of Rui L. Aguiar (PhD, Associated Professor with Aggregation at University of Aveiro) and Francisco Fontes (PhD, Auxiliar Professor at University of Aveiro)

o júri / the jury

presidente / president

José Luís Guimarães Oliveira
Professor Associado da Universidade de Aveiro (por delegação do Reitor da Universidade de Aveiro)

vogais / examiners committee

Rui Luís Andrade Aguiar
Professor Associado com Agregação da Universidade de Aveiro (orientador)

Francisco Manuel Marques Fontes
Professor Auxiliar da Universidade de Aveiro (co-orientador)

Manuel Alberto Pereira Ricardo
Professor Associado da Faculdade de Engenharia da Universidade do Porto

agradecimentos / acknowledgements

É com imenso gosto que chego ao fim desta etapa e tenho oportunidade para agradecer a quem mais contribuiu para este feito.

Agradeço em primeiro lugar ao meu professor e orientador Rui Aguiar pelo acompanhamento e pelo conhecimento que me transmitiu durante todo o tempo que estive no Instituto de Telecomunicações (IT), mesmo antes de iniciar esta dissertação.

Ao *Advanced Telecommunications and Networks Group* do IT e a toda a gente que, mesmo não sendo do grupo, estiveram do meu lado e de alguma forma contribuíram para este projecto. Um obrigado especial ao Rui Abreu Ferreira por todo o tempo, paciência, latim, e tinta vermelha que gastou comigo e com esta dissertação. O sentido de humor das suas correcções davam sempre outro ânimo ao meu trabalho.

A todos os meus amigos de Aveiro, em especial aos que mais estimo e mantenho próximos. É impossível agradecer o suficiente por todo o apoio e amizade com que preencheram estes anos de curso.

Agradeço especialmente à minha família pela força e todas as oportunidades que me conseguiram dar. Tentei aproveitá-las ao máximo e ser motivo de orgulho para vocês. Paulo, Isabel, Fábio, António, Elvira, José e Fernanda, esta dissertação é vossa. Deixo também o meu obrigado à família da minha namorada, por toda a ajuda e energia que me transmitiu.

Por fim, agradeço e dedico também esta dissertação à minha metade, Filipa Lopes. O apoio foi interminável, e por mais longe que estivesse, nunca me faltou suporte (e desafios) da sua parte.

palavras-chave proximidade, mobilidade, segurança, privacidade

resumo

A crescente adopção de dispositivos móveis, com cada vez mais capacidades de computação e comunicação, leva inevitavelmente à questão de como podem ser explorados. O objectivo desta dissertação passa por explorar algumas dessas capacidades de forma a melhorar e evoluir a interacção segura entre o utilizador e os serviços que utilizada no seu dia-a-dia. É particularmente interessante o uso destes dispositivos não apenas como sistemas de armazenamento, mas como peças activas na interacção entre o utilizador e o mundo que o rodeia, um cenário potenciado pelas crescentes capacidades de comunicação em proximidade destes dispositivos.

Esta dissertação debruça-se sobre o estudo e possível integração da proximidade física entre um utilizador e os sistemas que usa diariamente como um requisito extra na autenticação e comunicação entre eles, usando o seu dispositivo móvel para interagir com os mesmos. De forma a demonstrar uma possível integração destes elementos num sistema, este trabalho apresenta uma implementação que explora o uso de tecnologias de curto alcance como meio de comunicação e como requisito de autenticação, recorrendo a mecanismos de segurança para estabelecer comunicações privadas sobre redes públicas e garantir e verificar a autencidade da informação trocada e armazenada.

keywords proximity, mobility, security, privacy

abstract

The increasing adoption of mobile devices with more computing and communication capabilities inevitably raises the question of how to explore them. The goal of this dissertation is to explore some of those capabilities to improve and evolve secure interactions between the user and the services that he uses in his daily life. It is particularly interesting to use these devices not only as storage systems, but also as active elements in the interaction between the user and the world around him: this objective is boosted by the increasing proximity-based communication capabilities of those devices.

This dissertation focus on the study and possible integration of the physical proximity between a user and the systems he uses every day as an extra requirement for authentication, using his mobile device to interact with them. To demonstrate a possible integration of these elements into a system, this work presents an implementation that explores the use of short-range wireless technologies as a communication mean and as a requirement for authentication, using security mechanisms to establish private communications through public networks and to ensure and verify the authenticity of the information exchanged and stored.

Contents

List of Figures

Acronyms

ACO Authenticated Ciphering Offset. 11

AMP Alternative MAC/PHY. 8

AP Access Point. 15

BKS Bouncy Castle Keystore. 45

CA Certificate Authority. 22

CAM Continuous Active Mode. 14, 15

CCMP Counter Mode with Cipher Block Chaining Message Authentication Code Protocol. 15

EDR Enhanced Data Rate. 8, 10

GnuPG Gnu Privacy Guard. 22

IdM Identity Management. 32

IdP Identity Provider. 29

IrDA Infrared Data Association. 16, 17

IrLAP Infrared Link Access Protocol. 16

IrLMP Infrared Link Management Protocol. 16

IrPHY Infrared Physical Layer Specification. 16

ISM Industrial, Scientific and Medical. 8, 14, 19, 53

JKS Java Key Store. 45

L2CAP Logical Link Control and Adaptation Protocol. 10, 54

LM-IAS Link Management Information Access Service. 16

LM-MUX Link Management Multiplexer. 16

LMP Link Management Protocol. 10

LR-WPAN Low-Rate Wireless Personal Area Network. 19

NFC Near Field Communication. 18, 19, 27, 32, 33, 53, 54, 60

NWK Network Layer. 20

PAN Personal Area Network. 7, 16

PDU Protocol Data Unit. 12

PGP Pretty Good Privacy. 22

PSM Power Saving Mode. 14, 15

QR Code Quick Response Code. 41, 42, 46–48, 50, 55

REST Representational State Transfer. 40

RFCOMM Radio Frequency Communication. 40–42

RFID Radio-Frequency Identification. 17–19, 32, 33, 60

RSN Robust Security Network. 15

SDP Service Discovery Protocol. 12, 13, 46, 48

SIG Special Interest Group. 8, 9

SPKI . 22

SSL Secure Sockets Layer. 23, 37, 40, 46, 50, 54

SSP Simple Sharing Pairing. 10

TIM Traffic Indication Map. 15

TKIP Temporal Key Integrity Protocol. 15

TLS Transport Layer Security. 23–25, 40, 42, 54, 55

WEP Wired Equivalent Privacy. 15

WoT Web of Trust. 22

WPA Wi-Fi Protected Access. 15

Chapter 1

Introduction

Since the invention of the personal computer around the 70s, an increasing number of aspects of our lives went digital. Cellphones, smartphones, and more recently tablets, in particular, are seriously extending our lives with digital expressions of ourselves and others. Digital services are becoming more and more important, and are starting to replace traditional (physical) services in the lives of several people.

Whether for professional or personal reasons, we have extended our lives using electronic devices and digital services, eventually creating virtual representations of ourselves and our actions. These representations, or *digital identities*, are becoming more and more important as our online activity is increasing (through social networks, for example). In addition, these digital life aspects are starting to follow us wherever we go. With the increasing number of feature-phones and smartphones, being "always on" is now a reality, and our digital identities are increasingly present in these devices.

Sooner or later, our "digital selves" will be our primary way of identification and authentication, considering the logistic costs involved, extensibility and adaptability of digital systems, and enhanced security of this kind of information. ID cards, debit and credit cards, Facebook and Twitter accounts, and many more virtual identities will be available through our personal mobile devices, whether it is a smartphone, a tablet, or even a netbook. Most of our lives will have a digital counterpart, and these personal devices will become a true extension of each individual. There are already concepts for digital wallets [1] and trends are for these to become a privileged form of authentication.

These digital representations of ourselves represent the digital aspects concerning the mediation of people experience of their own identity and the identity of other people and things. A digital subject has a finite number of digital attributes (e.g., username, email address, etc.), and doesn't necessarily represent a human being (devices, services and other resources could be represented as digital subjects, for example). One of the main points for digital identity management is authentication. This is a key aspect of any kind of transaction, due to the trust that is necessary to establish. In order to prove its identity, a subject (whether a person or not) may need to present proofs of the veracity of its identity, like presenting a unique object (a key, for example), providing confidential information (like a password), proving the ownership of a personal resource (like an e-mail address, or even a digital token), or use a more robust and complex solution.

Authentication is crucial for some digital service, like online shopping, for example. Upon proper authentication, a user is able to buy and sell goods in a fast and very practical way. But there is a drawback in the use of such digital services. In order to pay online orders, for example, you have authenticate yourself, you have to provide private and sensitive data in order to prove your identity or your card's validity. This information, in the hands of ill intended people, can lead to numerous problems for their real owner, like identity theft or credit card fraud. These are just some examples of real and very serious problems that computer security faces, given the current volume of digital transactions. In order to secure digital services and maintain the users' trust, it is necessary to be one step ahead of attackers, improving the existing forms of defence and elaborating new ones.

A secure and simple way for the user to manage his digital representations is required to successfully migrate from simple physical tokens of identities to complex digital entities. Furthermore, the very privileged way of communicating with users and their devices will be upgraded, taking advantage of wireless technologies currently present on mobile devices. However, this transition entails some drawbacks, as it will be explained later. So, in order to overcome these drawbacks, the physical closeness between a user's device and the system with whom it is interacting is a characteristic worth exploring, since a wide range of situations where a person is required to prove its identity are performed in close proximity with the point of interest in question. This proximity interaction might be an interesting consideration to have in mind during the actual design of access control systems. This concept of *proximity* is the main focus of this dissertation, and it will be explained and developed throughout this work, in order to assess its potential as an asset to future access control systems. Bearing in mind that the client side is the primal ground of this dissertation, by gathering and studying the current state of the concepts and issues presented, it was possible to combine the gathered insight into an application that enables a mobile device to be used as a point of interaction with access control systems using tokens for authorisation purposes.

1.1 Goals

The proximity-based approach that this dissertation takes to access control aims primarily to enable new and feature-rich authentication scenarios and to effectively make them useful to the common user. Furthermore, it is important to provide the users with a simple and user-friendly way to manage digital identities and tokens they might possess, empowering them with total control over their information by turning their mobile devices into their personal digital wallets. Therefore, the overall purpose of this dissertation is to address the problem of secure, simple, and user-friendly access control management, exploring a user's mobile device to act as his authentication tool and, most importantly, to study the effects of adding proximity requirements to system-user interactions and verify what benefits it brings.

Consequently, the main goals of this dissertation include:

- The study of short-range wireless technologies, exploring the user proximity required in communications as an asset to the process;
- The study and design of a system to support a secure user authentication based on the closeness to personal mobile devices of the users;

- The modelling and implementation of an application able to capacitate a mobile device as a secure and easy to use proximity-based authentication mechanism, using digital tokens to provide proper authentication and authorisation.
- To provide a vision of the future work on the approached subjects.

1.2 Contributions

Most of the work of this dissertation was carried out under the *Multipass* project, a project funded by PT Inovação that focused on generating added-value for current Identity Management (IdM) architectures. It aimed to create new use cases and deployment scenarios that connect IdM with real world technologies, in order to pave the way for the integration of IdM and the Internet of Things (IoT).

The increasing use of pervasive technologies in our daily lives brings more and more possibilities of enriching and enhancing daily tasks and interactions. This project aimed to take advantage of these possibilities in order to bring advanced interactions to portable devices, making them more than just transport mediums of digital information. These devices would be an electronic extension of their users, allowing them to interact with their reality, using the advantages of IdM systems.

1.3 Structure

The remainder of this work is structured as follows:

Chapter 2 provides an insight of contemporary technologies and work being developed on the main areas covered in this dissertation, contextualising the reader. First of all, the two most prominent and pervasive concepts in this work are presented and properly explained. After that, some interesting and useful scenarios are formulated, involving access control through proximity-based user authentication, along with three service examples involving some of the formulated services. Then, the main security challenges of such services are identified and discussed. Finally, a detailed characterisation of the main current proximity technologies is also presented, along with other relevant technologies for the work developed in this dissertation.

Chapter 3 explains the details behind the architecture of a proximity-based service. The chapter provides necessary requirements of a system providing a service like the described in the previous chapter, the guidelines to consider during its design, and a description and explanation of instantiation details, such as entities and their interactions.

Chapter 4 is centred on the client side, the main focus of this dissertation. This chapter clarifies the choice of the mobile platform where the application was built, and several details about implementation. It also presents a functional evaluation of the developed application and the several tests it was submitted.

Chapter 5 finalises the dissertation with a conclusion about the developed work and its results, along with a forecast of the future work in the main issues covered by this dissertation.

Chapter 2

State of the Art

The ever growing presence of electronic devices in our society bring us countless possibilites of creating new uses and services that take advantage of their capabilities. As our daily actions are performed more and more using the technological environment around us, our digital presence becomes gradually more important and significant in our life. This massification of digital interaction ends up creating virtual representations of ourselves, digital identities that ultimately represent us whithin our digital parts of life. With the ubiquity of portable devices like smartphones, our digital identities are always close to us, thereby promoting their use and consequent interaction with other digital entities (whether they are people, services, or other resources with a digital representation). These digital identities play an important role in authentication, since they are used to establish trust between parties before performing any important interaction.

However, these digital counterparts of ourselves are not free from danger. Considering the amount of personal information they contain, the users' privacy must be ensured to provide a safe interaction environment and to avoid information theft. This involves secure information storing and, more importantly, safe data exchange between devices (since the user interacts through a mobile device, wireless communications are involved). The concept of proximity emerges when we are dealing with wireless interactions performed between two close participants, like a user and a system in his vicinity. If it proves an asset, it could lead to modern proximity-based systems, where people use their personal mobile devices to interact with systems close to them and to manage their digital identities.

The following sections of this chapter identify and clarify the several aspects related to this topic and its multidisciplinary. First of all, the main concepts to be aware of while reading this document are presented and explained to the reader, taking into account the scope of the dissertation. Then, the most relevant technologies for the implementation of this work are presented and analysed, in order to identify their strengths and weaknesses. After that, possible relevant services are formulated and their importance to our digital future is highlighted. Finally, the security challenges that might rise from future architectures and the use of digital identities are explored, in order to find the most secure and strong solution possible.

2.1 Concepts

All the work developed in this dissertation revolves around two key concepts that shape its objectives, requirements and results: *proximity* and *security*. To correctly address and use these notions throughout this document, we must first establish what they mean within the scope of this work.

2.1.1 Security

In general, *security* can be considered as the quality or state of being free from dangers. Computationally speaking, security has a narrow meaning, usually referring to protecting information from theft, corruption and other hazards, while keeping it intact and available to its intended users. According to this view, computer security refers to the processes and mechanisms used to protect information from improper publication, tampering or destruction by unauthorised and/or untrustworthy parties and unplanned events.

Despite of all available techniques and mechanisms to secure information, there are always attacks that can be performed to try to breach the protections applied. And given unlimited resources and time, attacks are considered to successfully overcome any security measures. Of course unlimited resources and time are commodities that are not available, but there are organisations and entities which indeed possess fair amounts of computational and human resources, and with enough time, they might be able to overcome most defensive measures.

The underlying costs of security measures within a system must also be considered. To design or adapt a system to include such defences is costly, either in money or in time, and adds extra complexity to any system. Security measures should be adapted to the sensitivity and importance of the information or system to guard, so no problems appear due to lack or excess of security [2].

Hence, a golden rule is usually taken into account when deciding which security measures to integrate into a system: information is considered secure as long as its lifetime (i.e. the time span in which the information is useful) is shorter that the expected time to overcome the security measures and get access to it. Note that the estimated time to overcome some computer defences depend mainly on the attacker's specifications. Concluding, the security aspects mentioned throughout this dissertation were build around this definition and considering the characteristics that it entails.

Mutual Authentication

Mutual authentication, sometimes referred as Two-Way authentication, plays an important role in the establishment of a trust relationship between a service and its user, contributing for an increase in the overall security of a transaction. In this process, both entities (the client and the server) try to authenticate each other in order to establish a secure and trustful connection between them. This way, the user knows for sure it is accessing the correct (and trustworthy) service, and the service insures it is only accessible to legitimate users.

Generally, there are three methods of authentication [3]:

What You Know Users are authenticated by proving the knowledge of some private information, difficult for other to have. User-Password based authentication is a good example of this scheme. These credentials cannot be stolen (from your mind), but only ensure that someone knows the information.

What You Have Users are authenticated by presenting an object like a passport, smart card, key, etc. The security of this authentication scheme relies on the difficulty of forging or acquiring such objects (they can be stolen or lost).

What You Are Users are authenticated by analysing their physical or behavioural characteristics. This is referred as Biometrics, and, comparing with the previous two, these credentials cannot be lost or forgotten, and most of them (DNA, iris, fingerprints) cannot be forged (although voice tones, for example, can be stolen, i.e. recorded). Behaviometrics is a term coined by some researchers, concerning the behaviour of a person (typing rhythm, locomotion, and speech, for example).

2.1.2 Proximity

According to the *Merrian-Webster Online Dictionary*, proximity can be defined as "*the quality or state of being proximate*". Indeed, this notion concerns the closeness between parties. Particularly, in this work, between a service provider (a server and its end-points) and the corresponding service consumer (a person using his personal mobile device).

So, why proximity? What does it bring or improve in a system? Where can it be used? What are its downsides? These are some important questions that need to be answered in full so proximity could be an important component to include as a feature of future systems.

The first key aspect of proximity is most obvious, to be near to the point of interest. This leads to a smaller action area, an area where the user and service providers have some kind of control. A smaller area means a smaller diversity of possible security threats to consider. It is simple to understand that, for instance, in an area of 1 square meter, it is not practical to have someone with a notebook and an antenna sniffing packets without being noticed. If you could restrict the information wirelessly sent to a small radius, it would be less likely to have that information reach vile hands.

To be close to the end point where we will authenticate ourselves ensures our presence, or at least the presence of the security token (or other mechanism) used to prove our identity. In actions that involve physical presence of the user, like going to a concert or pay our purchases at the local supermarket, proximity can be an asset, improving trust in transactions.

The use of proximity as a system key feature has been explored, specially for security reasons. In [4], proximity is seen as an important part of the system's security, targeted to mobile commerce (M-Commerce). The key feature of the system is the presence of a physically small device, like a ring, providing its presence to a security module. In [5], proximity between two devices is verified by capturing and comparing the relative similarities in wireless media of each device (like packet receptions, idle channel time, etc.).

In some cases, proximity gains a greater importance, since some activities make more sense when we are next to the devices that we are interacting with. Inspired by this idea, [6] describes a location-based authentication mechanism to prevent unauthorised access to a device's contents. For example, this system could force the user to be in the kitchen in order to turn on the stove.

2.2 Services

Some of the previous technologies are already present in everyday objects, like smartphones. Taking into account the non-stopping evolution of these devices and the constant increase of their capabilities, it is understandable their growing importance and indispensability in our lives. This way, it is inevitable the fact they are one of the privileged ways of interacting with whom (family and friends, for example) or what (e.g. websites and social networks) is not near us.

What if, taking advantage of the proximity and security technologies presented, personal mobile devices like smartphones could also be a way of communicating with people and specially systems around us, in our very neighborhood? These interactions would enrich the way we deal with our daily lives; with the right services, our smartphone would become our house keys, our IDs, our credit cards, our own wallet, and much more. These proximity services would be totally safe and very practical to use.

Thus, following this increasing trend, some examples of the most relevant services for a daily use are presented and described below.

Digital Lock

The digital lock is one of the first scenarios where access control could be applied, given the simplicity, importance, and ubiquity of the act of opening or closing a door. Using a personal mobile device as a "key" to operate an electronic lock may provide an improved way of controlling access to buildings. It is quite easy to picture a possible scenario involving digital locks[1]. A user's mobile device would possess some kind of private information to act as key, like a private key or even a certificate. This digital key would be completely private, unable to be shared with anyone and only present in its user's personal mobile device.

Nevertheless, the mere presence of the key should not be criteria enough to unlock the door. Otherwise, the user could open the door unconsciously, just by being around the door lock; or worst, just by stealing the device, some other person would easily open the door by approaching the device to the door lock. To avoid accidental openings of the door, it should be necessary for the user to provide additional information in order to successfully unlock his house, in the form of a challenge. Entering the correct PIN would successfully conclude the user authentication, unlocking the front door and allowing him to enter his house. With this system, there is the possibility of providing temporary access to our house to someone of trust, in a simple and controlled way. Just like the credentials stored in our device, a user could generate and provide credentials to someone of his trust to be able to enter his house on his absence. The user would have total control over the temporal validity of the credentials, being able to impose an expiration date or a maximum number of uses.

For added security and key management, it could even be possible to disable and mark as invalid any digital key stored in a lost or stolen device, just by calling the entity responsible for the house security system of the user.

[1] https://lockitron.com/

Digital ID

Using a mobile device as your digital ID card is somewhat similar to the digital key service, as it also intends to prove an identity, although with different objectives. This digital ID would hold all the information a regular ID card might have, with the exception of not having a physical support, only existing digitally. This enables the aggregation of all identification cards into one single digital device, creating one single point of user authentication [7]. Although convenient, this centralization would entail higher risks and negative consequences when such devices fail and miss their purpose. To support such centralised systems, a few measures would be necessary to be designed and implemented in order to successfully deal with eventual failures and problems of portable devices.

Considering most user's information gathered in a single device, it is possible to formulate a new way of sharing information, hierarchical-alike. A user would have total control over his information, using this control, for example, to establish different levels of clearance to access his personal information. Users would be able to group their information according to importance or relevance to other people or entities.

This service can be related with a scenario where the access to a user's info varies according to proper clearance of the enquiring device. This measure could enhance the control of personal info, preventing any device from becoming an "open book", and allowing only trustworthy people to access a user's personal info. The device itself could even ask for user permission in order to successfully share information with whom is requesting it, clearly informing the user about what it is being requested.

E-Ticketing

Ticket services are plausible interested parties in changing from physical to digital support. A ticket could be digitally bound to a person, so it could not be stolen and unauthorizedly used by someone else, it would have to be present on the device of the bounded person. Tickets would be bought online with no need of picking them up or print them before the show, only being necessary to prove the correct ownership of the ticket when entering the show, using your mobile device. Delegate a previously bought ticket to a friend, for example, would be more secure, since it would be digitally signed by the previous owner, confirming its authenticity.

This possibility is much better than the traditional e-Ticketing systems nowadays, where a user buys a ticket online and has to print the purchased ticket (after a download of a *pdf* document containing the ticket, for example) or receives a SMS with a unique code to be presented at the entrance of the event. These examples offer virtually no security at all, enabling the use of the tickets by someone who eventually steals them.

Wireless Payments

Credit and debit cards would be replaced by wireless payment systems, via Near Field Communication (NFC), Bluetooth or even Wi-Fi. Even online financial accounts, like PayPal, would be able to be used to pay bills through a personal portable device. Instead of carrying around physical money, a person would only need his ever present smartphone to pay his everyday expenditures, from meals to clothes, for example. A person could have more than one bank card

in the device, and the device could give instant feedback about purchases and resulting account balance. An interesting service would be generating temporary credit cards from a master credit card, with defined spending limits. For example, a parent could be interested in creating and giving to each of his children a credit card for emergencies, all of them linked to his master credit card and with a maximum spending limit. If one of his children used his credit card, the money would be deducted in the credit card account of his parent.

A recent example of a service of this kind is Google Wallet [8] [9]. This service will debut in Nexus S smartphones, and will enable purchases using NFC technology to exchange information. Although it is too early to tell if this particular service will prevail, more and more interest is being shown around NFC and mobile payments.

Context Ads

If unassisted communication were possible between a user's mobile device and certain system, i.e. if no confirmation of the user was necessary for information exchange to happen, a targeted advertisement delivery system is easily formulated. When enquired by a device with the correct credentials, a user's personal mobile device would reveal the profile of its owner, and according to this information, the user would receive in his mobile device ads according to his preferences, which were stored in the system. The ads could be sent considering the user's position inside a public space, like a shopping. The stores where the user spent most of his time would be favoured compared to others, at least for that user.

2.2.1 Scenarios

Having identified some examples of the most relevant services to be futurely integrated in our daily lives, it is natural to come up with some scenarios that we consider the best examples of all the previously presented. Concentrating our efforts on specific scenarios also allows a more elaborate and focused scope of the work.

In general, these scenarios describe physical interactions, around a mobile device, which establish relationships between different devices in its neighborhood, to enhance the user's experience. To correctly model the scenarios, it is necessary to idealise a set of physical interactions with the environment. We try to identify, within the scenarios' backgrounds, possible actions to be improved and facilitated using a personal mobile device as an interaction point using proximity communications.

This way, two types of interactions require special focus:

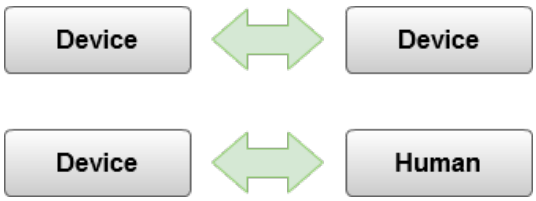


Figure 2.1: Interaction types.

Device to Device: Interactions between devices, or *Machine-to-Machine* (M2M). This kind of interaction is characterised by a normally non-assisted interaction (by the user). A complex set of information is exchanged to allow decision making by the devices. This context is important in a way that removes from the user some tiresome interactions with the surrounding environment, providing a better user experience.

Device to Human: Interactions between people and devices. This context is still important nowadays, since it allows to establish a close relationship between the user and the electronic systems that surround him. It also awares the user for the interactions to be performed.

The following scenarios occur involving the above two interaction types (Figure 2.1): Device to Human when a user interacts with his mobile device, and Device to Device when his mobile device interacts with surrounding systems.

Given these two preferred means on which the interactions will occur, it is essential to contextualise them into scenarios that can benefit from emerging technologies, such as new proximity technologies.

Scenario 1 - Digital Locks

This scenario describes the use of a device with short-range wireless communications to open a lock able to authenticate a user.

A person comes back home and, to open the door, places its mobile device near the lock. After detecting the device, the lock asks for credentials, and the user reveals its identity to the lock, to verify its access to the house.

The lock, after confirming that it is indeed the owner's "key", wanting to confirm too that it is the householder who is using the "key", requires such proof, asking the user to respond to a challenge posed by an operator/Identity Provider (IdP) (like typing a PIN), which unlocks the locks of the house. Given this, the lock opens, and the house just "recognised" the user as the householder, thus adapting to his presence, as far as possible, placing the office temperature as ideal, for example.

In this context, the mobile device behaves similarly to a conventional key, with the addiction of some features important to highlight. This device introduces an authentication process which requires, not only the correct key (stored in the device), but also a proof associated to the user's identity, in this cases represented by the user's PIN.

Since this is an access control mechanism, there is still the possibility that the householder delegates access to his home to other devices/users. For example, he may wish to delegate access to his home to someone he trusts, under special conditions, for example a neighbour during the holidays. This implies the ability to delegate capabilities to other devices other than their own. This delegation process, which in practice requires the ability to provide other devices with access tokens derived from those that already exist, may impose special restrictions that limits access based on a certain time limit or a number of uses.

It is also important to note that the authentication with the mobile device follows the same restrictions imposed to the authentication with the so called "traditional" devices, namely that the authentication process involves the conscious participation of individuals with a suitable

device, and the process cannot be completed without their consent. This should prevent, for example, the mobile device from inadvertently opening a door in close proximity of the user without his prior consent.

This scenario imposes certain requirements to the mobile device, including the ability to securely store information and to interact with its user in order to acquire additional information. Other requirements concern the way devices interacts with each other. Since this scenario enhances access control using mobile devices, safeguards should be introduced in order to ensure that, on one hand, the authentication system can not be subverted, and, on the other hand, information about the device's holder is not compromised. In other words, the authentication process is mutual. As a "door" (simple device that controls access to a location) will never grant access to a device with no credentials for such purpose, the device will also never initiate the authentication process towards a "gateway" to which it has no valid tokens.

Within this scenario, it is assumed that there is a prior record that associates the authentication tokens present on the household's device to the door of his house, being this relationship that allows the door to interact with him. It is also here that this way of controlling access to physical space offers advantages, since the costs of providing or revoke access to physical spaces to other devices are reduced.

This scenario incorporates the following concepts:

- Using a mobile device to gain access to an electronic lock.
- Mutual authentication between a personal mobile device and a access control device.
- Use of a (safe) digital key, to securely and privately interact with a "real" device adapted for that purpose.
- Using wireless communication, preferably a short range, to obtain credentials giving access to real-world barriers (adoption of the digital-to-real).
- Use of proximity communications to detect presence and enable local services (opening a door).
- Delegation of a digital key to access a physical space.

Scenario 2 - e-Ticket

This scenario describes how to use a personal mobile device to manage electronic tickets which give access to several resources.

To attend a concert of his favourite band, a user browses to its respective website, where he buys a access "key" or "ticket" to the concert hall. Arriving to the concert, he approaches his personal mobile device to the authenticator. After identifying the concert authenticator, the mobile device in question offers its user the possibility to activate his ticket. This process will grant him direct access to the entrance door, since verification of his key to that access control mechanism was correct, proving that the user possesses a valid ticket and satisfies all requirements to enter the concert. During this verification process, it can be necessary for the mobile device to provide additional attributes about the owner, for example, if the event requires a minimum age limit to enter.

There is also the possibility that, after purchasing the ticket, a user decides to offer it to a friend. In this case, the user must delegate the existing ticket in his device to his friend's device. The ticket delegation process allows a device to issue a new ticket from the original,

which indicates that the ticket has been delegated.

Besides concerts, a user can consult its itinerary through every museum, theatre, or any other show the user had been, since the information was securely stored in his mobile device.

In this scenario, the user's personal device acts like an access key to a resource, in this case an area associated to an event. Since this type of access control is normally associated with commercial transactions, it is important to ensure that the user's device maintain a copy of all the used credentials, as well as a record of their use. Therefore, it is expected for the ticket issuing process to digitally imprint in them important properties and information: the tickets are globally unique (identifying who issued them and the event that the ticket is concerned); contain information that allows mobile devices to identify valid authenticators for this ticket; and contains information that allow the verification of the ticket's validity.

With a process of mutual authentication between the device and the authenticator, the device holder can prove that he holds valid credentials, but also that he made use of them. Thus, neither party can repudiate the process. Having delegated the e-Ticket, supported by his mobile device, the properties of non-repudiation must remain intact.

This scenario incorporates the following concepts:

- Use of temporary and limited credentials for access to restricted places.
- Use of authentication mechanisms to protect mutual interactions.
- Use of location and wireless communication technologies, to enhance identity, identification, and interaction services.
- Delegation of a digital key to access an event.

Scenario 3 - Digital ID

This scenario describes how to use a personal mobile device that serves as identification, both visual and digital, to interact in different ways that require weak and strong authentication.

A student, intending to request a registration certificate, goes to the university's office. At the entrance, the access control system, with his permission and interacting with his personal mobile device, is able to identify him as a student of that institution (although no other information is revealed), which gives the security guard authorisation for letting him in. However, inside the building, to request the certificate, the student needs to authenticate himself furthermore, to verify informations such as his current academic year, for example. Using his mobile device, he willingly authenticates himself near an authentication device of the office, which displays the necessary extra information to the employee processing his request.

In this scenario, the student's personal mobile device works as a form of authentication between multiple individuals connected to the same entity (the University), but where each plays a distinct role. The device works as a storage medium for the credentials, verifies the authenticity of other authentication devices, and notifies its carrier before providing any of its information to third parties. During the scenario, the student's mobile device works as an authentication mechanism for services from other authentication devices, allowing each party to apply access control according to the identity of the others.

This scenario can be easily extended to support more backgrounds. For example, public entities could have direct access to your citizen information without the user's approval, given

their authority.

This scenario incorporates the following concepts:

- Interactions between two mobile devices with privacy control.
- Use of digital keys and credentials to different authorisation identity proofs.
- Use of mutual authentication mechanisms, supported by the devices.
- Use of proximity and wireless technologies, to obtain access to buildings controlled by people and to services provided by people in regular places.
- Use of digital technology to fulfil everyday interactions, safely, without breaking the established natural processes.

2.2.2 Security Challenges

The biggest threat this kind of system has to face is probably the theft of a user's mobile device. In a user perspective, it is the first issue to be solved in order to effectively provide the added security of a digital alternative for today's methods. If an ill intended person is able to steal a personal mobile device, it is imperative to have security measures that can prevent him from using that device or any information in it to harm its owner. Therefore, this solution needs security mechanisms able to protect, in case of emergency, any information contained in these devices.

To prevent harmful use of all the personal information a personal device might have, taking in account the presented scenarios (ID cards, tickets, and bank account information, for example), it is required to securely store them in the device, and if possible, even render them useless if compromised. Personal info could be encrypted when stored in the user device, and a master password could be requested in order to access the content of the information.

If the device in question could use SIM cards, it could be possible to bind every user information to the SIM keys, requiring the device to use the correct SIM card in order to access the desired data. The SIM operator could even render useless any information binded to the SIM card if its user flagged it as stolen, or even block the entire device to prevent its misuse.

To bypass the need for the personal device to store all information about the user, the use of an IdM back-end could be applied. This would require fewer information to be storer on the physical device, since most of it would be online, provided by the IdM service. Every operation, or just the important ones, would be performed with online access, in order to verify each party involved and to securely store important data. This feature also acts as backup and improves recovery in case of theft or damage of the device. A user would be able to authenticate himself in a new device in no time, being able to access all the services he had available in the previous device. No tickets would be unusable, no credit cards taken, and no contact info would be lost.

Along with theft, impersonation is another real concern to take in account. It is a very serious situation when someone manages to gather sensitive information about someone else. It is even more serious and potentially dangerous if such information is successfully used to trick other people and services to impersonate the person from whom the data was collected. Usually, this situation causes severe losses to the person who was "impersonated", whether financially, socially or legally.

Apart from what has been described, the fact that communications can be conducted in a

wireless way brings more threats to the equation, although it is a much more convenient way to exchange information. In a wireless network, information travels through the airwaves, not physical wires, so anyone within range can "listen in" the network, and worst, are able to "inject" packets of malicious data into the network. This possibility leads to common attacks like:

Spoofing In a spoofing attack, a person or program tries to masquerade itself as another by falsifying data, in order to gain an illegitimate advantage or access.

Eavesdropping Since anyone within range can capture any packet transmitted in a wireless network, it is possible to eavesdrop communications between parties. This is specially dangerous for unsecured connections, where messages are transmitted in plain text.

Brute-force Depending on the type of network being considered, brute force attacks can be a threat (and it is a problem that goes beyond wireless networks). Basically, this trivial and general problem-solving technique consists in systematically enumerating all possible solutions for a certain challenge and checking whether each candidate solution is a correct answer to the challenge.

Data Modification Technologies like Radio-Frequency Identification (RFID) and NFC are vulnerable to data modification attacks. Using an RFID jammer, it is possible to disturb the signal sent from a device. In a best case scenario, this attack is able to render useless any information sent from the target device.

Replay Attack Information captured by eavesdropping a wireless communication can be useful, in this case by replaying a captured message. If no message flow control is implemented, it may be possible for an attacker to replay previously captured messages and those messages being classified as valid by the receiver.

Man-in-the-Middle The attacker performs an active eavesdropping, establishing independent connections with the victims and replaying messages between them, making them believe that they are talking directly to each other over a private and secure channel, when in fact the whole conversation is being controlled and modified to the attacker's advantage. To perform this, the attacker must intercept all messages passed through the two parties.

Denial of Service A DoS attack is an attempt to make a resource unavailable to its users. Usually manifests itself as flood of requests, like connection attempts or content requests. The principle behind this attack is to overload the available resources of the target, in order to render them useless to perform any useful task by occupying them all.

2.3 Technologies

To effectively incorporate the necessary proximity and security within a system, the right set of tools must be used. For proximity sensing, the need is for short-range wireless technologies, since user-friendliness is desired (i.e. cordless exchange of information) and we seek to create a zone similar to a Personal Area Network (PAN) where communications will occur. For security, the primary need is for some kind of mechanism to enable secure and private communications between two peers. Even if eavesdropped, the messages caught must be impenetrable, preventing any unauthorised party from reading their contents.

This way, the most relevant technologies regarding the fields and concepts of this dissertation are detailed and analysed in this section, in order to understand the most beneficial ones for the stated goals.

2.3.1 IEEE 802.11

One of the most commonly used wireless protocols, IEEE 802.11, is a set of standards for wireless local area network computer communications. These standards are created and maintained by the IEEE LAN/MAN Standards Committee. 802.11 technology had its origins in a 1985 ruling by the U.S. Federal Communications Commission, which released several bands of the radio spectrum for unlicensed use. Many technology firms began building wireless networks and devices, taking advantage of this newly available radio spectrum, but devices from different manufacturers were rarely compatible, due to the lack of a common wireless standard. Eventually a committee of industry leaders came up with a common standard, which was approved in 1997 by the IEEE.

The 802.11 set consists of several protocols and amendments, each one bringing new features to wireless communications. 802.11-1997 was the first wireless networking standard, but 802.11b was the first widely accepted one, and more recently the 802.11g and 802.11n standards became widespread. 802.11b and 802.11g operate in the 2.4 GHz Industrial, Scientific and Medical (ISM) band, which occasionally may lead devices that use this standard to suffer interference from microwave ovens, cordless telephones, and Bluetooth devices. To control and minimise any eventual interference, 802.11 uses direct-sequence spread spectrum and orthogonal frequency-division multiplexing signalling methods.

802.11 was introduced in 1997, allowing connection speeds of 1 and 2 MBit/s, and indoor range of 20 meters and outdoor range of 100 meters. In 1999, 802.11a was introduced, operating in the 5 GHz band, achieving a maximum net data range of 54 MBit/s. Operating in the 5 GHz band gives a significant advantage, avoiding the heavily used 2.4 GHz band reduces potential interference from most of wireless devices. It is capable of achieving 35 meters indoor and 120 meters outdoor. This standard is nowadays mostly obsolete, due to its range limitation (walls and other obstructions severely affect 802.11 signals) and 802.11b popularity (802.11a products started being shipped later due to their additional manufacturing difficulty, thus contributing to the wide adoption of the less-expensive 802.11b products).

802.11b was also introduced in 1999, and brought a dramatic throughput increase (compared to the original standard) and substantial hardware price reduction. These two factors led to the rapid acceptance of 802.11b as the definitive wireless LAN technology. Since 8002.11b operates at a lower band than 802.11a (2.4 GHz), it is capable of (theoretically) achieving ranges of 40 meters indoors and 140 meters outdoors - operating at a lower carrier frequency minimises signal absorption by walls and other solid objects in their path, due to higher wavelength.

In June 2003, another standard was released, 802.11g. Like 802.11b, it works in the 2.4 GHz band, so it can suffer the same interference issues as 802.11b. Also just as 802.11b, it can (theoretically) achieve ranges of 40 meters indoors and 140 meters outdoors.

Finally, in 2009, IEEE approved the 802.11n amendment, which improves the previous 802.11 in many ways, such as adding multiple-input multiple-output antennas. Thanks to this, the wireless range was increased to 70 meters indoors, and 250 meters outdoors.

Regarding energy consumption, 802.11 features a rather simple power saving mechanism [10] [11]. An IEEE 802.11 based wireless network interface can choose to stay in one of the two states anytime, awake or asleep, more precisely, in Continuous Active Mode (CAM) or Power Saving Mode (PSM). In CAM, the radio is always powered up, and the wireless interface can perform data communications or stay idle. Instead, in PSM, the radio is turned off, making it impossible to detect or sense the network behaviour of others.

Wireless communications proceed as usual in CAM, with devices being able to sense network behaviour of others and receive data anytime; however, in PSM, Access Points (APs) buffer incoming frames destined for mobile stations in PSM, and periodically announce their buffering status through a Traffic Indication Map (TIM), contained in the beacon frame. Periodically, mobile stations wake up to listen to the beacon frames. In the unicast case, the mobile station initialise a PS-Poll frame to the AP to retrieve data, and the AP responds each poll with one buffered frame (multiple polls can be submitted until all frames have been retrieved). In the broadcast/multicast (B/M) case, the existence of buffered frames in the AP is indicated through the DTIM, which is a special TIM sent out at a fixed number of beacon intervals. After the DTIM transmission, all the B/M frames the AP buffered are delivered immediately.

Unlike the CAM, a mobile station in PSM normally has opportunities to turn its network interface off, saving energy when there is no data outstanding at the AP. The use of PSM can greatly reduce the energy consumption and extend the mobile stations' lifetime, in light to moderate traffic load [10].

Although this mechanism is able to save some energy in Wi-Fi interfaces, these interfaces drain up to 50% of the total energy a mobile device uses [12].

802.11 had originally weak security on purpose due to multi-governmental meddling on export requirements [13], but was later enhanced through the 802.11i amendment after governmental and legislative changes. In 1997, when 802.11 first appeared, Wired Equivalent Privacy (WEP) was introduced, since wireless transmissions are susceptible to eavesdropping. However, by 2001 many weaknesses have already been identified [14], leading to attacks able to intercept transmissions and gain unauthorised access to wireless networks. Nowadays, WEP is deprecated, since it can be easily cracked within minutes with specific software. In response to these vulnerabilities, the IEEE created the 802.11i task force. This amendment specifies security mechanisms for wireless networks. Although the Wi-Fi Alliance had previously introduced Wi-Fi Protected Access (WPA), it just implements a subset of 802.11i. The full implementation of the 802.11i by the Wi-Fi Alliance is known as WPA2, also called Robust Security Network (RSN). The 802.11i amendment provided a RSN with two new protocols, the 4-Way Handshake and the Group Key Handshake. This security network also provides two confidentiality and integrity protocols, Temporal Key Integrity Protocol (TKIP) and Counter Mode with Cipher Block Chaining Message Authentication Code Protocol (CCMP).

The current 802.11 hardware is quite small: there are chips small enough to include in mobile phones and other portable devices. 802.11 hotspots are something very common and abundant these days, which motivates the integration of this technology in personal portable devices. 802.11 chip prices are an important factor in its popularity, since they cost from 2 up to 8 euros [15] [16] (depending on the protocol - a, b, g or n). 802.11 implementations are extremely popular these days, having all sorts of devices with this technology and numerous wireless internet hotspots around the world.

2.3.2 Infrared

Infrared light is an electromagnetic radiation, having a wavelength between 0.7 and 300 micrometres and a frequency range between 1 and 430 THz. This kind of radiation is not visible, since its wavelength is longer than of visible light (about 390 to 750 nm) [17].

The Infrared Data Association (IrDA) defines communication protocol standards [18] for the exchange of data over infrared light. This type of communication is usual for PANs. IrDA is a short-range optical communication, where devices, in order to communicate, must have a direct line of sight to each other [19].

IrDA has several specifications. The mandatory Infrared Physical Layer Specification (IrPHY) is the lowest layer of the IrDA specifications, defining the range, angle, speed, modulation and wavelength of the communications. Infrared Link Access Protocol (IrLAP) is also mandatory, and is the second layer of the IrDA specifications. It provides guidelines for access control, device discovery, addressing conflict resolving, connection initiation, information exchange and connection termination. During data transfer, IrLAP is also responsible for providing reliable error detection, retransmission, and flow control. The mandatory Infrared Link Management Protocol (IrLMP) is the third layer of the IrDA specifications, and consists of two parts: Link Management Multiplexer (LM-MUX), which is responsible for providing multiplexed channel on top of an IrLAP connection, and Link Management Information Access Service (LM-IAS), which operates in a "client-server" manner, where service providers register their services so other devices can access them [20].

By default, IrDA defines 1 meter for maximum range between standard infrared devices [20], although it varies for communications involving low power profile devices (0.3 meters for communications between standard and low power, and 0.2 meters for communications between low power). Typically, IrDA communications work best from 5 to 60 cm away from the transceiver. Note that IrDA data communications operate in half-duplex mode, since a device's receiver is blinded by the light of its own transmitter while transmitting. So, with this technology, full-duplex communications are not possible.

IrDA has low power consumption, and there are several procedures and techniques for saving power. There is also a low power version, with less range (typically a maximum range of 30 cm) and consumes 10 times less power compared to the standard version [21].

IrDA does not contain any encryption or other means of security. However, it is usually considered secure because of its limited range and the line of sight need. In order to eavesdrop on a communication, it is necessary to be in the direct vicinity of the communicating devices and, on top of that, stand within the angle limitations (somewhere between 15 and 30 degrees [21], like in Figure 2.2).

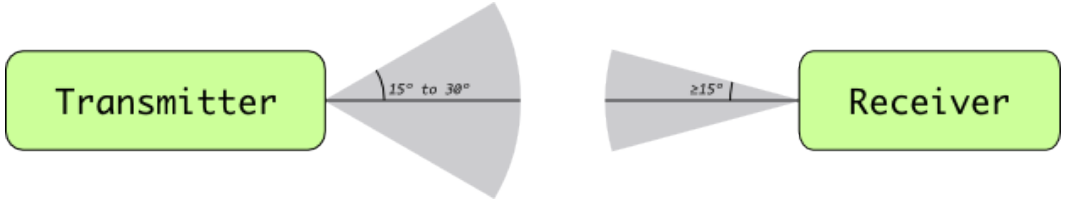


Figure 2.2: IrDA transmission angles.

Nowadays, IrDA adapters are very small and compact, and can easily be included in small devices like smartphones. Also, their prices are quite low, with IrDA chips costing less that 2 euros [19].

There is a variety of devices employing infrared light for wireless communications. The most common use is in television remote controls, but it is possible to find this technology in computers, PDAs and video game consoles. IrDA popularity reached its peak between the late 90s and the early 2000s. The emergence of other wireless technologies like Bluetooth and 802.11 (Wi-Fi, more precisely) overshadowed this technology, since they don't need a direct line of sight to exchange data. However, IrDA is making a comeback with highly efficient protocols, with higher speeds (IrSimple achieves at least 4 to 10 times faster data transmission speeds than existing protocols[2]) and other features, like the possibility to connect an infrared device to a local area network [22].

2.3.3 Bluetooth

Bluetooth was created as a wireless alternative to data cables, in order to connect multiple devices together (with none of its cluttering). It emerged in 1994 by Ericsson [23], and is currently managed by the Bluetooth Special Interest Group (SIG), which oversees the development and licensing of Bluetooth standards and technologies.

Bluetooth was named after the king Harald Blåtand (translated as *Bluetooth* in English), also known as Harald I of Denmark, who united rival Danish tribes into a single kingdom. Bluetooth technology was designed to do the same for devices, uniting them into a single universal standard for short-range wireless communications.

Since its first version (v1.0 specification), Bluetooth had its share of changes and improvements, specially because it had many problems in its early versions, like lack of anonymity and interoperability issues. Version 2.0 introduced an Enhanced Data Rate (EDR) mode for faster data transference, but it also provided lower power consumption through a reduced duty cycle. In version 2.1, another mode was added, called Sniff Subrating, which was designed to increase battery life for devices whose typical usage involves a significant amount of inactive time (like keyboards, mice, headsets). In version 3.0, the addiction of Alternative MAC/PHY (AMP) brought another energy consumption reduction. This feature uses a Bluetooth link for device discovery, initial connection and profile configuration, and after that phase, a 802.11 link for achieving (theoretically) data transfer speeds of up to 24 MBit/s. This means low power connection models of Bluetooth are used when the system is idle, and low power per bit radios are used when large quantities of data need to be sent, thereby enhancing its energy efficiency. Finally, in April, 2010, the Bluetooth SIG completed the Bluetooth Core Specification version 4.0, which includes the Bluetooth Low Energy protocol. This feature is an enhancement to the Bluetooth standard, allowing two types of implementation, dual-mode and single-mode. Dual-mode implementations have the Bluetooth low energy functionality integrated into traditional Bluetooth controllers (v2.1 and bellow). The resulting technology shares much of classic Bluetooth existing radio and functionality, leading to a minimal cost increase compared to classic Bluetooth technology. It is even possible to use current Bluetooth technology chips with this new low energy feature, allowing the development of classic Bluetooth enabled devices with new capabilities and features. Single-mode chips possess a lightweight Link Layer, which provides

[2] http://www.irda.org/displaycommon.cfm?an=1&subarticlenbr=48

ultra-low power consumption in idle mode operation, simple device discovery, and reliable point-to-multipoint data transfer with advanced power-save and secure encrypted connections. This low power improvement is expected to be included in fitness wearable devices, for example. It was specially designed for up to one year battery life devices, such as those powered by coin-cell batteries. An example of its use, would be Bluetooth enabled wristwatches, using the low energy stack to display the Caller ID information, or to monitor the wearer's heart rate during exercise, provided by Bluetooth low energy sensors in its clothes.

Bluetooth works in the globally unlicensed ISM radio band (2.4 GHz), dividing such band into 79 smaller channels of 1 MHz each. Communication devices using the ISM band must tolerate interferences from other ISM devices, so this band is usually used for unlicensed operation. However, not all wireless technologies are robust enough to manage or even negate disturbance effects from devices operating in the same band [24].

A master-slave structure is present in Bluetooth communications. A master device is able to communicate with up to 7 slave devices [18] (though not all devices support this limit, like most of the headsets), creating a piconet. All devices in a piconet share the master's clock, and packet exchange is based on such clock. The protocol core specification provides ways for two or more piconest to connect to each other, forming a scatternet. In this situation, certain devices simultaneously play the master role in one piconet, and the slave role in another. Devices are able to switch roles, by agreement.

Low power consumption was another original consideration of Bluetooth [24], especially to be easily integrated within mobile devices, for example. According to its specification, this is reinforced by allowing radios to be powered down when inactive. The most commonly used radio is Class 2 and uses 2.5 mW of power, although this number may vary - that's why there is no table of Bluetooth power consumption, its specification is so flexible and customisable that power consumption depends on the aspects used [25]. In [26], measurements show Bluetooth power consumption of a master device around 17.5 mA and a slave device around 31 mA.

Bluetooth was designed as a short-range wireless technology, providing three classes of range: Class 1, having a range of 100 meters; Class 2, having a range of 10 meters; and Class 3, having a range of up to 1 meter. With 3 different ranges, it is possible to choose the most suitable for the desired application. We must consider that restraining the wireless action range of an application or device might be important, since Bluetooth devices do not have to be in line of sight of each other to communicate, thus raising some security issues as previously stated.

Security was always an important aspect of Bluetooth, and is even more important nowadays. The Bluetooth SIG has a Security Expert Group, who provides critical security information and requirements as the specification evolves. Nevertheless, since its first version, vulnerabilities have been discovered and patched along the years, contributing to a more secure Bluetooth specification and reducing the number of possible attacks. Even so, Bluetooth security continues to evolve, since the complexity of malicious attacks continues to increase. Natively, Bluetooth has several features and techniques to make it difficult to intercept, or eavesdrop transmissions and its contents [27]. Its specification includes security features at the link level [28], featuring single or mutual authentication, encryption, and a frequency-hopping spread spectrum technique to send and receive messages [29]. Each Bluetooth device features a 48-bit address, theoretically unique for each device (2^{48} possible numbers), defined by the IEEE, and has three mechanisms for maintaining security at the link level: private authentication key, a 128-bit random number used for authentication purposes; private encryption key, from 8 up to 128 bits in length, used

for encryption; and a 128-bit frequently changing random number, generated by the device itself. The Bluetooth authentication scheme uses a challenge-response strategy, through a two-move protocol to check if the other party knows the secret key.

Currently, Bluetooth technology is prone to several security issues and threats [30] [31]. Although some of them were fixed throughout the versions, there are still a few present in the latest versions. Some vulnerabilities are only found in certain implementations (i.e., wrong or faulty implementation by manufactures), and some security issues are characteristic of the Bluetooth specifications. Nevertheless, Bluetooth SIG keeps working to remove any vulnerability in this technology and ensuring futures devices improved security.

The various versions of Bluetooth specifications define four security modes, and every Bluetooth device must operate in one of them. Each version supports some, but not all, of the four modes.

Security Mode 1: Non-secure mode. No security is implemented, so no authentication and encryption are applied, leaving connections and the device itself vulnerable to attackers. This mode is only supported in v2.0 + EDR (and earlier) devices.

Security Mode 2: Service-level security mode. Security procedures are initiated after Link Management Protocol (LMP) link establishment but before the Logical Link Control and Adaptation Protocol (L2CAP) channel establishment. In this mode, a security manager maintains policies to control access to specific services and devices. It is possible to grant access to some services, without providing access to other services.

Security Mode 3: Link-level security mode. In this mode, the security procedures are initialised before the physical link is fully established. Authentication (unidirectional and mutual) and encryption are supported in this mode, and mandates authentication and encryption for all connections to and from the device.

Security Mode 4: Service-level security mode. Similar to Security Mode 2, but introduces the Simple Sharing Pairing (SSP). This mode is mandatory for communications between v2.1 + EDR devices, and if the other device does not support this mode, the Security Mode 2 is to be used.

Bluetooth only supports challenge-response based authentication schemes [29]. Each authentication process is composed by the *claimant*, the device attempting to prove its identity, and the *verifier*, the device validating the claimant's identify. The validation of the devices is accomplished by verifying the knowledge of a secret key - the Bluetooth link key.

The authentication occurs as follows (Figure 2.3):

1. The verifier sends the claimant a 128-bit random number to be authenticated.
2. Both participants use the E_1 algorithm[3] with the random number, the link key, and its 48-bit Bluetooth device address as inputs, to compute an authentication response. The 32 most significant bits of the response (SRES) are used for authentication purposes, and the remaining 96 bits are saved as the Authenticated Ciphering Offset (ACO) value (later used to create the Bluetooth encryption key).

[3]The E_1 algorithm is based on SAFER+. SAFER algorithms are iterated block ciphers (IBC), in which the same cryptographic function is applied for a specified number of rounds.

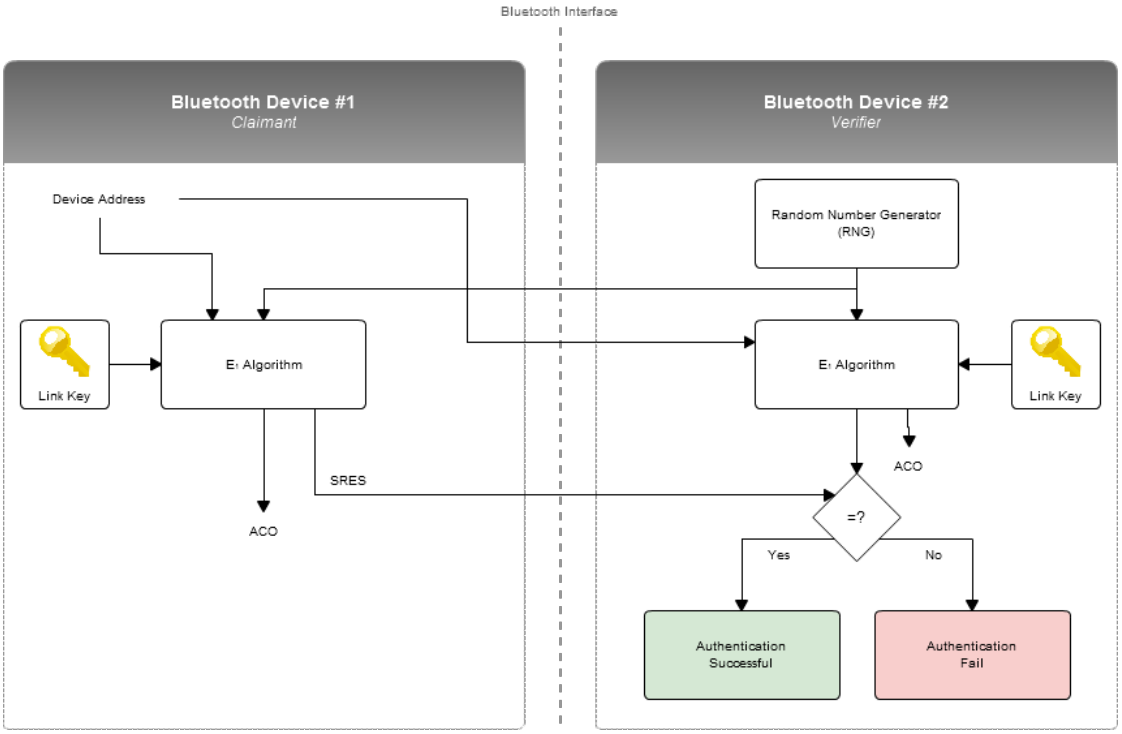


Figure 2.3: Diagram of Bluetooth Authentication.

3. The claimant sends the 32 most significant bits of the computed response to the verifier.
4. The verifier compares the received SRES from the claimant with the value it computed.
5. If both values are equal, the verifier considers the authentication successful; otherwise, the authentication fails. Each time the authentication fails, there is a period of waiting time until a new attempt can be made, which doubles for each subsequent failed attempt from the same address (until the maximum waiting time is reached). When no failed attempts are made during a time period, the waiting time decreases exponentially (up to a minimum waiting time).

Mutual authentication is also included in the Bluetooth standard. In order to set up a mutual authenticated connection between two devices, the above process just needs to be repeated in the same two devices which already established a connection, just by having the claimant and the verifier switching roles.

Alongside with the Security Modes, Bluetooth supports an additional confidentiality service to resist eavesdropping attempts on the packets exchanged in Bluetooth connections. It features two encryption modes to provide confidentiality and a third unencrypted mode (Table 2.1).

Besides all this, Bluetooth still allows two levels of trust, related with the three levels of service security [30]: a **trusted device** possesses a fixed relationship with another device, thus having full access to services, and an **untrusted device** does not have a fixed relationship with other Bluetooth devices, leading to restricted access to services. Three levels were defined for

Table 2.1: Bluetooth encryption modes.

Mode	Description
1	No encryption is performed on any traffic
2	Unicast (single addressed) traffic is encrypted using encryption keys based on individual link keys; Broadcast traffic is not encrypted
3	All traffic to all devices is encrypted, using an encryption key based on the master key

Bluetooth services, stating the requirements for authorisation, authentication and encryption, as seen in Table 2.2:

Table 2.2: Bluetooth security levels.

Security Level	Description
1	Authorisation and authentication is required. Only trusted devices are granted access. Untrusted devices need manual authorisation
2	Only authentication is required. An application is only granted access after an authentication procedure.
3	No special requirement, open to all devices. Automatically granted access.

Bluetooth features rather small and cheap hardware, since chip sizes are around a couple of centimetres wide, smaller than a coin, and their prices are usually bellow 1 euro, partially due to its popularity on mobile devices. Bluetooth has almost 18 years of development and is quite reliable, thus justifying the large number of devices worldwide, believed to be more than 700 million.

Bluetooth Service Discovery Protocol

One of the features that differentiates Bluetooth from other short-range wireless technologies is the inclusion of a protocol stack to service discovery, called Service Discovery Protocol (SDP). This feature allows devices to discover services advertised and supported by other devices.

SDP resorts to a request/response model (Figure 2.4), with each transaction consisting of one request Protocol Data Unit (PDU) and one response PDU. Bluetooth use L2CAP transport protocol, and only one SDP request PDU per connection to a given SDP server may be outstanding at a given instant. I. e., a client must receive a response to each request before issuing another request on the same L2CAP connection. This limitation provides a simple form of flow control.

A Bluetooth enabled device has all the information about the services its SDP possesses within a service record, which is a list of service attributes. Each service attribute describes a single characteristic of a service. A service attribute consists of the following two components:

Attribute ID 16-bit unsigned integer that distinguishes each service attribute within a service

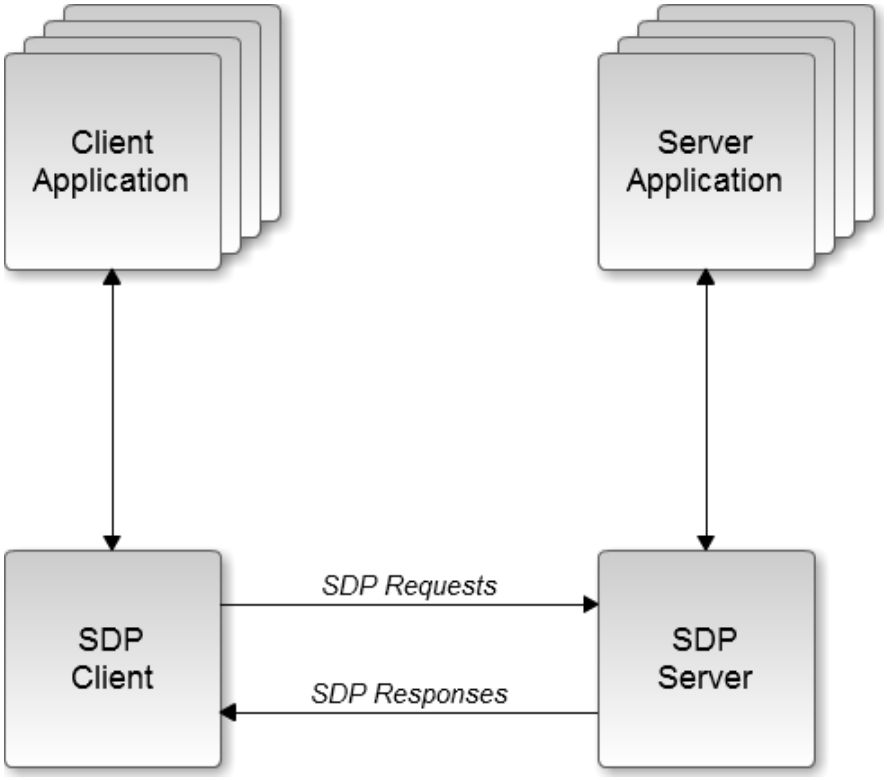


Figure 2.4: Bluetooth SDP connection diagram.

record. This attribute also identifies the semantics of the associated attribute value.

Attribute Value Variable length field determined by its associated attribute ID and by the service class of the service record in which the attribute is contained. An attribute value is represented as a data element, which is a typed data representation consisting of the following two fields:

Header The header field is composed of two parts. A Type Descriptor, which is a 5-bit descriptor of the data element type, containing the 5 most significant bits of the first byte of the data element header; and a Size Descriptor, which is a 3-bit data element size index followed by 0, 8, 16 or 32 bits, containing the least significant 3 bits of the first byte of the data element header.

Data A sequence of bytes with a length specified in the size descriptor and a meaning (partially) specified by the type descriptor.

Besides the previous characteristics, each service is an instance of a service class. The parent class definition provides the definitions of all attributes contained in service records representing instances of that class. Each attribute definition states the attribute ID, and the intended use and format of the attribute value. Each service class possesses a unique identifier, represented as a UUID. The Universally Unique Identifier is supposed to be unique across all space and time (128-bit value).

Bluetooth SDP allows two means to discover what services other Bluetooth devices are offering:

Searching This feature allows devices to retrieve the service record handles for particular service records, based on the attribute values stored in those service records. It is only possible to search for attributes whose values are UUIDs. To locate the desired service, service search patterns are used, which are basically a list of UUIDs (services attributes) used to locate matching service records.

Browsing This mechanism is based on an attribute shared by all service classes, called the *BrowseGroupList* attribute. Its value contains a list of UUIDs, each on representing a browse group with which a service may be associated for browsing purposes. To browse a SDP server's services, a device creates a service search pattern containing the UUID that represents the root browse group. All browsable services are made members of the root browse group by having the root browse groupâĂŹs UUID as a value within the BrowseGroupList attribute.

2.3.4 ZigBee

ZigBee is a set of specifications for a high level communication standard, enabling low-cost, low-power consumption, two way, wireless communication. It is based on the IEEE 802.15.4 standard for Low-Rate Wireless Personal Area Networks (LR-WPANs). The relationship between IEEE 802.15.4 and this technology is defined by 802.15.4 stating the structure and guidelines of the Physical and Medium Access Layer of the protocol, while ZigBee focuses the Network and Application Layer stack [32]. ZigBee operates in the ISM radio bands, more specifically, 868 MHz in Europe, 915 MHz in the USA and Australia, and 2.4 GHz worldwide [33] (basic bit rate of 250 kbit/s for this frequency [34], lower data rates for the previous).

While Bluetooth devices just need to have the mandatory core functions, and additional features are implemented only if desired by its manufactures, ZigBee devices must implement every specification of the protocol. ZigBee protocol stack is usually around 40kB, and unlike Bluetooth, which supports only 8 devices in a piconet (1 master and 7 slaves), it supports up to 2^{16} devices in the same network [34]. One of the best features of ZigBee, and a valuable one for long wireless networks, is its self-healing ability. If a network node becomes offline for some reason, the network automatically changes its architecture if necessary.

Around 1998, networks similar to ZigBee began to be conceived, since it was realised Wi-Fi and Bluetooth were not suitable for many applications. In 2003, the IEEE 802.15.4-2003 was completed, later superseded by the publication of IEEE 802.15.4-2006. The ZigBee Alliance was formed in October 2002, and ratified ZigBee 1.0 specification on 14 December 2004, announcing public availability of its specification on 13 June 2005. The ZigBee Alliance is an association of companies (consisting of leading semiconductor manufacturers, technology providers, original equipment manufacturers and end-users) with common interest in enabling reliable, low-power, and cost-effective global standards for monitoring and control products.

There are three node types in a ZigBee self-organised network [32]:

- Coordinator: The main head of the network, being typically mains powered. Has the unique function of forming the network, and is responsible for establishing the operating channel and the PAN ID. Once the network is formed, the Coordinator functions like a router device.

- ZigBee Router: Full function devices, mains or battery powered (preferentially mains). Creates and maintains information about the network and uses it to determine the best route for a data packet. In order to allow other Routers or End devices to join in, it must join a network first. Can be a data packet source, and can route data packets to and from other nodes.
- End device: Reduced function devices, always battery powered. Must always interact with its 'parents' in order to receive or transmit data, and it cannot route traffic.

A ZigBee network consists of one Coordinator and one or more Routers and/or End devices [34]. The ZigBee Network Layer (NWK) supports star, tree and mesh topologies. A single device controls the star topology (the Coordinator), and is responsible for initialising and maintaining the devices in the network. In the tree and mesh topologies, the Coordinator takes responsibility for initiating the network and choosing certain key network options. Tree networks move data and control messages using a hierarchical routing strategy, while mesh topology allows full peer-to-peer behaviour (each router is usually connected through at least two pathways).

ZigBee communication range varies from implementation to implementation. The IEEE 802.15.4 does not specify range requirements, since its objective was to define both the physical and data-link layers for providing low data rate and long battery life, with low complexity. Usually, transmission range varies from 10 to 75 meters [35][34], but it is easily reduced or increased depending on the adapters used.

One of ZigBee's important features is its power consumption and, consequently, battery life. Usually, ZigBee devices' battery life is around years [36], unlike Bluetooth, which just lasts for some days. A light switch, for instance, with around 6 operations per day (on/off), is able to achieve a lifetime of 10 years, using a 3V LiMn coin cell. It all depends of the system's power-saving modes and battery-optimised network parameters of the devices, such as a selection of beacon intervals, guaranteed time slots, and enablement/disablement options[4].

ZigBee security incorporates the strong security elements of 802.15.4, implementing two extra security layers on top of them: Network and Application security layers. Its simple, yet strong, security features rely on the 128-bits AES encryption algorithm [37]. Its specifications provides:

- Sequential freshness: using an ordered sequence of inputs, it is possible to identify and reject replayed frames, thus preventing replay attacks.
- Frame integrity checking functions: a message integrity code (MIC) is used to protect against data modification from parties without the cryptographic key, thus ensuring that data came from a device with the cryptographic key.
- Entity authentication service: using a shared key, this service provides secure means for information synchronisation and authenticity between devices.
- Data encryption: using a symmetric cipher (shared key between two or more peers), ZigBee devices are able to encrypt data to protect it from being read by anyone without the cryptographic key.
- Trust Center: all ZigBee networks must have only one Trust Center, which is responsible for deciding whether to allow or disallow new devices into its network. It may periodically change the network key, broadcasting first the new key encrypted with the old one. This role is usually performed by the Coordinator, though it is possible to have a dedicated

[4] http://www.meshnetics.com/zigbee-faq/#14

device for it.

ZigBee chips are very small, incorporating a programmable microprocessor, RF radio, network protocol stack, and memory in a 7x7mm wide microchip. These microchips can cost from 2 to around 15 euros, depending on the quantities and the manufacturer [38] [39] [40].

2.3.5 RFID

RFID is a generic term used to describe the technology involved in a system that wireless transmits the identity of an object or a person using radio waves. This technology is theoretically similar to barcode identification, but based on communications via electromagnetic waves, which make it possible to communicate without direct line of sight [41]. It is usually used for identification and tracking of objects and people.

Basic RFID systems normally involve two components [42]:

1. An interrogator, also known as reader, that communicates with the RFID tags through radio waves;
2. A tag, that stores unique information about the object or person where it was applied.

It is possible to be considered as a third part of the system the software used to manage the reader and the information it receives [43]. Most RFID tags consist of at least two parts. An integrated circuit, responsible for storing and processing information, modulating and demodulating radio-frequency signals, and other functions; and an antenna, for receiving and transmitting radio-frequency signals.

There are three types of RFID tags [44]:

1. Passive RFID tags, which possess no power source and require an external electromagnetic field to start signal transmissions;

2. Active RFID tags, containing a battery so it can transmit signals as soon an external source has been identified, without the need of being powered up by it;

3. Battery assisted passive (BAP) RFID tags, which require a radio signal from an external source to wake up, but are capable of providing greater communication range.

The first true use of radio identification similar to nowadays RFID was in 1973, when Mario Cardullo created a passive radio transponder with memory [45]. Since it was passive, it had to be powered by the interrogating signal (similar to passive RFID tags).

There are several established RFID standards [46], and a few emerging. They deal with the air interface protocol (how tags and readers communicate), data content (how data is formatted and organised), conformance (how to make sure products meet the standard), and applications (how the standards are used). Since there is no global public entity governing the used frequencies for RFID and stipulating standards for this technology, in principle, every country and/or continent can set its own regulations. This leads to a lack of device classes regarding its features, unlike Bluetooth or 802.11, which created many different implementations of this technology and with ranges from a few centimetres up to around 20 meters.

The maximum operating range of RFID varies between 10 cm and 1m, according to its standards [47]. However, the actual range of RFID systems is able to vary widely, because it is dependent on antenna design and size, system power, transponder power consumption, and receiver sensivity [48]. Active RFID tags can, nevetheless, be operated from 100m or even more, thank to the power supply they possess.

Regarding power consumption, the use of passive tags results in devices which need no energy to communicate by radio, but they can not start transmissions, just respond to them (it is necessary to receive an external signal to be powered up, as previously mentioned). Also, passive tags are not usually used in devices, and they are usually integrated in objects or even animals. In order to actively communicate, RFID devices need a power source, so it can be able to initiate communications with tags (becoming a reader).

The simple design applied in RFID tags makes them unable to support typical security mechanisms. However, new generation tags may be capable of symmetric-key cryptography, thus offering a native security function to protect communications [42]. For now, most RFID tags are possible to be copied (i.e., since most RFID tags are used as replacements for optical barcodes, their information can easily be retrived just by queryring them by radio waves), even important documents like modern passports [49] [50], for example.

RFID tags are very cheap nowadays, due to their simplicity (they require no power sorce or advance circuit). Each passive tag costs costs around 5 cents, and active tags are around 25 cents [51]. Readers range from 350 to 1400 euros, since readers are typically purchased as a part of a complete system, including software for example.

2.3.6 NFC

NFC is a short-range wireless technology created to enable data exchange between devices. This technology evolved from a combination of existing technologies (Proximity Cards and RFID), it extends the ISO/IEC 14443 standard [52], enabling all NFC devices to communicate with existing ISO/IEC 14443 smart cards and readers and other NFC devices.

NFC was designed as a short-range wireless technology, with a maximum working distance of 20 cm (but typically only 3 centimetres [52]) hence making it a suitable technology to use in usually crowded environments. It works in the 13.56 MHz carrier frequency and at rates of 106, 212, 424 and recently 848 kbit/s.

NFC technology has low power requirements (non optimised prototypes are able to use only 30 mA of power [53], and more developed solutions are able to reach less than 15 mA), similarly to Bluetooth v4.0 Low Energy protocol. However, the NFC power consumption is greater than Bluetooth v4.0 Low Energy protocol when reading from an unpowered device, since extra energy is necessary to activate the passive tag (an average consumption of 30 mA).

While its small action range might decrease attack possibilities, this characteristic alone can not ensure secure communications. Unfortunately, NFC does not provide any native mechanism to protect communications, being vulnerable to eavesdropping and data modification, for example. In order to securely exchange data through NFC, it is necessary for applications to use higher-layer security protocols to establish secure channels.

The lack of link level security in NFC technology was explored by Haselsteiner and Breitfuß in [54], allowing them to perform attacks from eavesdropping to data modification, insertion,

corruption and Man-in-the-Middle. Rieback et al. demonstrated in [55] a SQL injection attack as well as self-replicating RFID viruses. In [56], a study is conducted in which several possible attacks against NFC-enabled devices are identified, like URI Spoofing or rebooting the device's GUI. However, it is necessary to consider that some of these attacks are only possible due to faulty implementations on the device.

Like RFID, NFC modules can be very small, fitting inside 40mm x 21mm x 5mm modules (including antenna). Chip prices are around 4 euros, making NFC an affordable option for wireless communication.

NFC is primarily aimed at mobile usage, and it is used mainly for three specific purposes:

1. Reader: the NFC device reads passive RFID tags, such as contact informations or advertisement;
2. Card emulation: the NFC device simulates a existing card, like a ticket or a credit card (similar to proximity cards);
3. P2P: two NFC devices communicate with each other to exchange information.

Practical applications of these uses are:

- Mobile ticketing, for transports, museums, cinemas, and similar settings.
- Mobile payments, like a debit card for small purchases.
- Enhanced reality, by reading RFID information from outdoor billboards for example.
- Identity management, to use a NFC device like a Identity Card.
- Electronic Key, replacing physical keys of houses, cars, hotel rooms, and other locks.

Numerous trials of systems using NFC technology are being made worldwide, specially related to mobile payments [57] [58]. Usability studies about NFC and contactless payments reported acceptance and appreciation by its participants regarding the concept of incorporating information transfer and secure payment functionality into mobile phones.

2.3.7 QR Codes

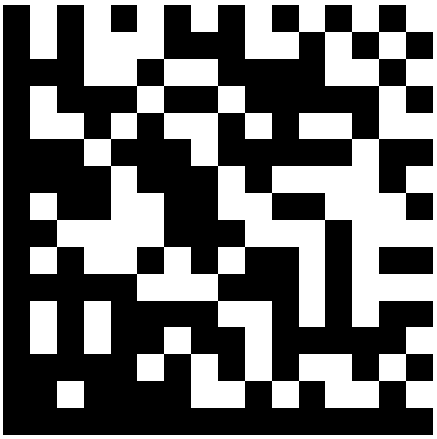
Barcodes are also a proximity technology, usually requiring close proximity in order to scan the symbols. They require line of sight, and barcode scanners usually must be just a couple of centimeters away to successfully read a barcode. Currently, Quick Response Codes (QR Codes) are a type of barcode quite common and popular worldwide, so their study was included in this dissertation.

As barcodes, QR Codes are optical machine-readable representations of data, containing data about the entity to which they are attached. Barcodes originally used the variation of width and spacing of parallel lines to represent data (linear or 1 dimensional barcodes), but currently 2 dimensional (2D) barcodes use triangles, rectangles, dots and other geometric patterns.

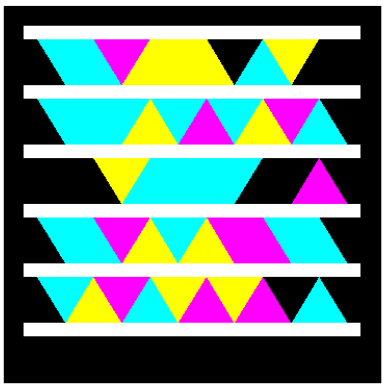
Linear barcodes can only encode numbers, and until recently, sophisticated devices were required for decoding. The evolution of image processing and multimedia features of mobile devices adds barcode encoding and decoding capabilities to these devices. Moreover, barcodes usually encode a unique serial number that is a key into a database containing further information, but the need to encode more information wanted the barcode itself to be a portable database, which lead to 2D codes.

The first truly two-dimentional barcode was introduced in 1988, and since then, several other formats and standards have been presented. 2D codes store information along the height and width of the symbol, removing the vertical redundancy of linear barcodes. Thus, 2D barcodes are able to store much more data than linear barcode. Regular barcodes have a capacity of 10 to 22 characters, while a QR Code, for example, can hold up to 7089 digits, 4296 letters or 2953 binary data [59]. 2D symbols are also highly reliable and durable, even if 50% of them have been vandalized [60].

The recent inclusion of cameras in mobile phones and their increasing image quality was a decisive fact in the current popularity of 2D barcodes. They can be easily printed in magazines and newspapers, for example, or used to label products and goods. They can even be transmitted electronically.



(a) Data Matrix



(b) High Capacity Color Barcode (HCCB)

(c) Quick Response Code (QR Code)

Figure 2.5: Two-dimensional barcode examples.

There are several 2D barcode formats available for use, but only the 3 more currently relevant formats will be addressed.

Datra Matrix (Figure 2.5(a)) was invented in the late 1980s, and it is currently one of the most used 2D barcodes. It is recognised by the International Standards Organisation, and is heavily used in aerospace, electronic and automotive industries to label components and documents [61]. Every Data Matrix is composed by an "L" shape in the lower left corner (a *finder pattern* to locate and orient the symbol) and two other borders consisting of alternating dark and light small squares (a *timing pattern* to provide a count of the number of the symbol's rows and columns).

High Capacity Color Barcode (HCCB) (Figure 2.5(b)) is a technology developed by Microsoft which uses clusters of colored triangles, instead of the regular rectangles of 2D barcodes. A palette of 4 or 8 colours is used to achieve the high capacity, but it also supports the use of black and white when necessary. Laboratory tests using 8 colors have yielded 3500 alphabetical characters per square inch [62], using standard off-the-shelf printers and scanners.

QR Code (Figure 2.5(c)) is another format of 2D barcodes, created in 1994. Initially used for tracking parts in vehicle manufactoring, they are now used in much broader context, given their current popularity. Similarly to Data Matrix, it is composed of positioning/aligning and timing elements (among others, like version and format information, and error correction keys), and it uses dark and light rectangles to represent data. There has been a great interest in this format nowadays, especially in mobile-oriented applications. Android operating system contributed to the current popularity of QR Codes, using this format to send metadata to existing applications on Android mobile devices, like opening a browser with a link scanned from a QR Code or adding a person's contacts after scanning a QR Code with his vCard.

2.3.8 Public-key Cryptography

Public-key cryptography is a term regarding a set of methods designed to encode information using asymmetric keys and asymmetric key algorithms (i.e., the information used to transform the message to a secure form - public key - is different from the information used to reverse the process - private key). Basically, these algorithms create a mathematically related key pair, containing a public key (available to anyone) and a private key (secret to everyone except its owner). Unlike symmetric key algorithms, which use identical cryptographic keys for encryption and decryption, asymmetric algorithms generate two different, yet mathematically related, keys (although related, the private key cannot be derived from the public key) [63].

Public-key cryptosystems, besides encryption, are also used for digital signing, in which a message signed by the sender's private key can be verified by anyone with access to the sender's public key, hence proving the sender's access to the private key, and therefore is likely that he is the associated person of that key pair and that the message has not been tampered with [63].

Compared to symmetric key algorithms of equivalent security, public key algorithms known so far are relatively computationally costly (due to the use of typically larger keys). This way, hybrid cryptosystems are used for practical efficiency reasons [64]. In such systems, one party generates a shared secret key (a session key, much shorter than a traditional public key) which is then encrypted by each recipient's public key. Each recipient, using the corresponding private key, decrypts the session key. Once all parties know the session key, they can start using it to encrypt and decrypt messages, through a much faster symmetric algorithm. Some hybrid schemes use a session key unique to each message exchange.

Although these Public-key schemes are considered quite secure, they are still susceptible to

brute force key search attacks, systematically checking all possible keys until the correct one is found. For example, if a public-key algorithm is used to encrypt a session key, and an attacker can generate a database of all possible session keys encrypted with a certain public key, it is much easier for the attacker to look for a match in the database than trying to break the public key.

Public-key algorithms are designed to resist chosen-plain text attacks, and its security depends on the difficulty of deducting the secret (private) key from the public key and the difficulty of deducting the plain text from the cipher text. Basically, the complexity of breaking a key will grow along with the size of the key. However, most public-key algorithms are vulnerable to chosen-cipher text attacks.

In order to correctly bind public keys to their respective owners, a Public-key Infrastructure is typically used. A Registration Authority (RA) assures the correct binding between a user identity and its keys (which, depending on the level of assurance the binding requests, may be carried out by software, or under human supervision), and then a Certificate Authority (CA) approves the request. Using its own key, the CA authenticates the given user's key, so that everyone who trusts the CA will also trust the given user. Currently, several vendors provide public keys and bindings to user identities, being specially used to encrypt and/or authenticate e-mail messages, documents and applications, and to establish secure communications.

Apart from CAs, there are two alternative approaches to collect this trust: (SPKI) and Web of Trust (WoT).

Simple Public-key Infrastructure In SPKI, the authorisation is integral to its design. SPKI can be used to provide simple and effective access control, binding a user's authority with a public key. A SPKI certificate could be used, for example, to easily authorise a subject to read a certain file until a determinate date, having the authorisation been signed by an issuer, using its private key [65]. The importance of SPKI is in the separation of authentication and authorisation, improving privacy (no ID is necessary in authorisations, and it can not be inferred from the certificates) [66].

Web of Trust This decentralised model uses self-signed certificates and third party verifications of them. Pretty Good Privacy (PGP) and Gnu Privacy Guard (GnuPG) are two implementations of this model. Using the words of PGP creator, Phil Zimmermann:

> *" As time goes on, you will accumulate keys from other people that you may want to designate as trusted introducers. Everyone else will each choose their own trusted introducers. And everyone will gradually accumulate and distribute with their key a collection of certifying signatures from other people, with the expectation that anyone receiving it will trust at least one or two of the signatures. This will cause the emergence of a decentralised fault-tolerant web of confidence for all public keys. "*

In a Web of Trust, each system user is able to choose for himself who to trust or not. If it is not possible to personally confirm if the subject S owns the private key of a certain public key, it is possible to look for users that already certified the subject S as trusted. This flexible model allows the existence of many independent webs of trust throughout computer networks, and any user can be a part of, and a link between, multiple webs, using their identity certificate.

2.3.9 Transport Layer Security

Transport Layer Security (TLS) and its predecessor, Secure Sockets Layer (SSL), are high level cryptographic protocols designed to provide secure communications between parties (preventing eavesdropping and tampering), mostly over the Internet. Web servers and web browsers usually rely on the SSL protocol to create encrypted channels for private communications over public networks.

TLS is the successor of Secure Sockets Layer (SSL), which appeared in 1995 (version 2.0, since version 1.0 was never publicly released) by Netscape. In 1996 version 3.0 was released, fixing some security flaws. TLS 1.0 (SSL 3.1) was defined in 1999 as an upgrade to SSL (version 3.0), although these protocols do not interoperate due to their differences (TLS does include means to downgrade to a SSL 3.0 connection). In 2006, TLS 1.1 (SSL 3.2) was defined and in 2008, TLS 1.2 (SSL 3.3) was released.

A stateful connection is negotiated between a TLS client and server in a handshaking procedure, where some connection parameters are agreed between them in order to successfully establish a connection. A typical TLS session follows the following steps (summarized in Figure 2.6):

1. A client sends a message to a server, specifying the highest TLS protocol version it supports, a list of suggested cipher suites and compression methods, and a random number. If the client is attempting a session resumption, he may send a session ID too.

2. The server responds, indicating the chosen protocol, cipher suite and compression method (from the choices offered by the client), and a random number. The server should always choose the highest version both itself and the client support.

3. The server then sends its certificate to the client (this step might be omitted, depending on the selected cipher), and signals the end of the handshake negotiation by sending a *ServerHelloDone* message.

4. The client responds with a message containing a *PreMasterSecret*, public key, or nothing (depending on the selected cipher).

5. From the random numbers and the *PreMasterSecret*, both client and server compute a common secret (the "master secret"), from where all other key data for the connection is derived.

6. After the handshake, the client sends a *ChangeCipherSpec* message, declaring the authentication (and, if negotiated, the encryption) of its messages from now on. Finally, the client sends a *Finished* message, containing a hash and a message authentication code (MAC) over the previous messages. The server will attempt to decrypt this message, in order to verify the hash and the MAC.

7. Now, the server sends a similar message, a *ChangeCipherSpec* as mentioned above, and completes the handshake with a Finished message too. The client tries to perform the same decryption and verification as the server.

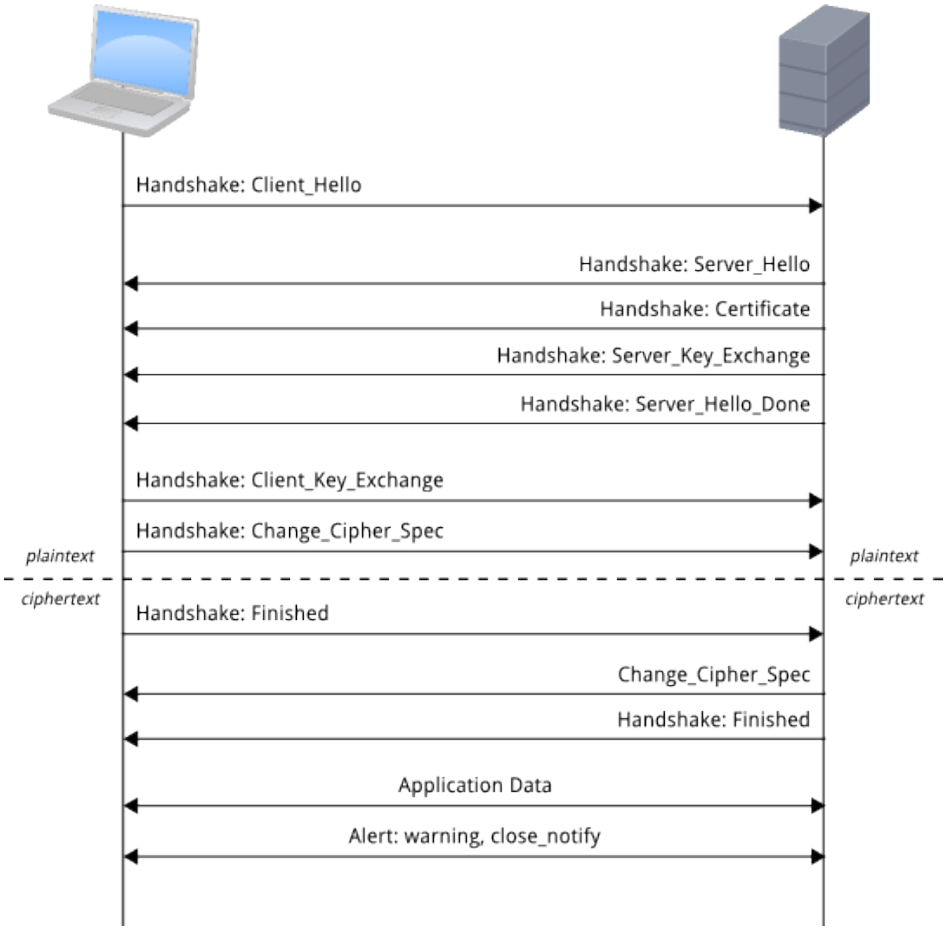


Figure 2.6: Sequence diagram of a TLS connection establishment.

Mutual Authentication for TLS

A TLS connection can be set to require mutual authentication by enabling Client-authenticated TLS Handshakes. This feature allows authentication and non-repudiation of the client, using digital certificates.

A TLS connection requiring Client Authentication varies on the following steps:

1. After the server has sent his certificate to the client, it requests a certificate from the client using a *CertificateRequest* message.

2. The client, before sending the *PreMasterSecret* and so on, sends its certificate to the server.

3. After the *ClientKeyExchange* message, the client sends a *CertificateVerify* message, a signature over the previous handshake messages. It can be verified using the client's certificate's public key, thus showing the server that the client has access to the private key of the certificate and possesses the certificate.

2.4 Summary

Considering all the studied short-range wireless protocols, the chosen technology to carry out the proximity communications with the personal mobile devices was Bluetooth. The main reasons behind this choice were its security features and wide availability. As previously mentioned, Bluetooth incorporates several features to ensure secure communications, like frequency hopping, device pairing and PIN request. These features remove several security issues from the application layer, thus facilitating the development of secure communications. Bluetooth technology is also extremely popular, being present in most of modern portable devices, from cellphones to tablets. The ubiquity Bluetooth has achieved ensures connectivity between almost every kind of device, and most important, every current smartphone features Bluetooth connectivity, making them possible wireless authentication devices. With all of this popularity, Bluetooth support is extensive, and its continued development is ensured.

However, NFC is getting a lot of attention these days, and although the number of NFC-enabled smartphones available is very limited, forecasts [67] point out to more than 500 million NFC phones worldwide. The interoperability with most RFID equipment, the short interaction range, and the low connection establishment latency are three of the most important characteristics of NFC, and they make it a very practical and interesting technology to include in our lives. Considering NFC possibilities and future availability, it might be a technology to work side by side with Bluetooth in smartphones or even replace it. Future uses for NFC technology would include health monitor devices, keycards to lock/unlock doors, exchange information between devices, or even pair Bluetooth devices just by tapping them together. NFC is intended to be simple and easy to use, providing instant interactions just by tapping devices.

In terms of scenarios, we focused in the e-Ticketing setting due to its flexibility and completeness. This is a very complete setting in terms of technology and interaction aspects to explore, and proposes a rather plausible practical scenario, justifiable in a near future. It also proposes not only the technological aspects, but also their applicability to a well-known problem.

Chapter 3

Architecture for Proximity Services

To experiment and demonstrate the concepts discussed in Chapter 2, a prototype covering some key aspects of this dissertation was designed and built. This prototype was developed based on a scenario chosen from those presented in chapter 2: the e-Ticketing setting. However, the system architecture of the prototype is very generic when it comes to proximity-based services, and it can be applied to other scenarios as well, given its flexibility.

The prototype will simulate an e-Ticketing system, were a user is able to buy tickets for shows, which he can turn in afterwards when accessing the show. The tickets are available for purchase online; after any ticket purchase, the user downloads the corresponding ticket. When approaching the show's entry point, the user is able to turn in the previously purchased ticket, using proximity communication to perform the data exchange; if he possesses the correct ticket, the system will grant him access to the show.

The architecture of the prototype implemented for this dissertation is presented in this section, along with a detailed description of the project requirements, guidelines established and followed throughout the whole process, and its later instantiation.

3.1 Requirements

For a product or prototype development, it is necessary to gather the proper requirements. Requirements are documented needs of what a particular product or service should be or perform. They identify necessary attributes, capabilities, or qualities a particular system must have in order for it to have some value of use by people or other systems.

First of all, the requirements of our target architecture must be found, analysing the goals previously set and the scenario in hand.

3.1.1 Proximity

As mentioned before, one of the main cornerstones of this dissertation is the explanation of proximity as an asset to access control systems, so its integration into the system is mandatory. The requirement of having the personal mobile device in the vicinity of the system to interact with will be accomplished using short-range technologies. These require a distance of a few

centimetres to only a few meters between the two communication points (depending on the technology used and its settings), and by controlling this distance it is possible to set some kind of interaction space around devices, in which both devices are sensing and looking for peers to interact with.

To meet this requirement, both the user's device and the developed system must possess a short-range wireless adapter of the same technology. Current smartphones have no problem with this, but the prototype must include an interaction end-point or module responsible for performing the communications between it and the user's device.

3.1.2 Security

Probably the major concern about the architecture of proximity services and the interactions between modules within it is security, specially the security of communications. When migrating physical, tangible actions or procedures to a digital medium, the security and privacy of such operations should meet high security standards, since the users are transferring the trust they have in those manual and physical actions to a new medium, with specific issues being hard for them to grasp.

The security referred must be applied both in the storage and communication of data. Personal mobile devices must be able to safely store their own information. In case of misappropriation or compromise of a user's device, this ability might stop vile hands from retrieving important and sensitive data from stolen or compromised devices. They must also be capable of communicating with other devices in a secure and confidential way, providing defences against possible eavesdropping. Since the communication takes place in a cordless way, extra risks come from it, and thus the necessity of transmitting only secured data.

3.1.3 Ticket Handling

The prototype will simulate an e-Ticketing solution, so it must support ticket handling, like checking, buying, and turning in tickets. These tickets will act as unique access tokens which upon proper validation grant the user with access to the required show.

The ticket selling point provides a website displaying all the available shows users are able to buy tickets to. When a user buys a tickets, the server must be capable of generating a digital ticket to deliver to the user, containing metadata about the show and important cryptographyc data for the user to safely establish a connection with the entry point when turning in the ticket and to prove its authenticity.

A user would be able to buy tickets with his mobile device, accessing some kind of online ticket store. There, he would be able to browse for shows and check their details, and after choosing the desired show, he would be able to buy the corresponding ticket and proceed to its download.

Upon ticket purchase, the user could see the list of purchased tickets in his mobile device, along with checking each ticket details for more detailed information. Tickets should be saved in the user's device, in a transparent yet secure way.

On arrival at a show for which the user has a valid ticket, he uses his personal mobile device to

communicate with the access control system in the spot responsible for access control. Then, he searchs for the correct service and ticket, and after some data exchange, if the ticket is considered valid and corresponds to the show, the system grants the user access to enter.

3.2 Guidelines

A guideline is a recommended practice that allows some leeway in its interpretation, implementation, or use, but aims to guide its followers to a common goal or desired practice.

Software development is mindless when no guidelines are followed, leading to future and inconvenient problems. The lack of lines of thought to correctly guide the development scatters the developers and negatively affects the final product. Thus, this section points out and explain the key guidelines followed in this prototype.

Designing a proper prototype to demonstrate the benefits of proximity-based systems must follow special guidelines, considering, for example, the mobile environment where the user interacts or the proximity requirement. Thus, this section points out and explain the key guidelines followed in this architecture, and that should guide all prototypes.

3.2.1 Modularity

A modular architecture consists in dividing a system into smaller units, preferably independent among themselves, serving different purposes inside the system. Such designs offers several advantages, like separation of responsibilities (each unit having its own responsibility) and extensibility (allowing an easy later extension of the system virtually without changing it). This extensibility allows future addition of features to the system, simply by creating new modules and coupling them into the system without significant adjustments. This line of though was had in mind in this prototype's design, in order to create an infrastructure capable of being reused for future proximity-based systems.

3.2.2 Issuing first, Consuming after

It is important to underline the need for the architecture to support the separation between the purchase and the delivery. The credential creation and its later consumption must be executed in separate phases to allow interactions with different providers and to provide greater flexibility for the user.

This guideline is important for a modular and robust system, which allows extra freedom when it comes to the user's action. The purchase and later delivery of tickets can be performed by two complete different systems from two different entities, having only in common a specific expected ticket structure. The user might even be able to delegate his ticket to another person, acting like a intermediate selling point. This separation between the ticket issuing and downloading and the subsequent ticket delivery makes the system much more flexible and user-friendly.

3.2.3 Mutual Authentication

Although security mechanisms are applied in this design, there is no impregnable way of protecting digital information and communications. Some of the threats to be concerned of are Man-in-the-Middle attacks. To render useless such attacks, a mutual authentication approach was followed in this work.

In the e-Ticketing scenario chosen for the prototype, an ill intended person, with the right info and tools, might be able to forge a ticket in order for the system to grant him access for a show for which he does not have a valid ticket. Such possibility can be countered by imposing mutual authentication during ticket delivery. During the establishment of the SSL connection with the service platform for turning in the ticket, in addition to the server having to present the correct credentials to the client (thus proving it is a trustworthy entry point), the user's device must present valid credentials too, proving the ticket is authentic and hence not forged.

3.2.4 User considerations

Although the system environment explained so far involves important and rather complex security notions for the common user, the client application should be easily handled and understandable by almost any user. Furthermore, the user should be able to grasp any interaction involving his device. The client application should inform the user about any action it is performing and request, if necessary, the user's permission in order to allow interactions with his device.

Likewise, to avoid requiring the user to be familiar with advanced security concepts, most of the security requirements are not user dependent, which means less responsibility and effort by the user. This leads the application to provide only high-level feedback to the user, like if he intends to present the selected ticket to a certain service, for example.

3.2.5 Security vs. Usability

Balancing security and usability in authentication methods is a complex subject. Two requirements must be considered: keep ill intended people from unauthorisedly accessing a user's account and information, and allow that user to access his account and information. Both aspects are equally important, and a fine balance must be achieved for a system to be secure and usable. Too little security and information gets deleted/stolen/duplicated and much more; too much security and the system will be unusable (or too frustrating and annoying to be used) [2]. So a middle ground between security and usability must be settled in order to design an usable and interesting application for the user.

3.2.6 "Smart" Connectivity

Wireless connectivity should not be "always on", should be only turned on when a user intends to initiate a connection. This reduces some hazards, like spoofing or eavesdropping, and reduces eventual interferences between devices. When the action the user chose in the application requires interaction, if the adapter is not enabled, the application informs the user of

the wireless needs, and asks him if he want to turn the adapter on. After the required interaction is performed, the application turns the adapter off.

3.2.7 User's Awareness

Although most of the security processes are responsibility of the server-side of the system, one of the objectives to have in mind is to empower the user with the right information, properly informing the user about the information being requested from him, the interaction attempts with his personal device, and the underlying risks of the interactions. The user has the final word about the information that leaves his device, and which interactions to trust. Above all, he must be conscious about who or what (in case of a device-device interaction) is requesting information or any sort of action approval, in order to protect his privacy.

3.3 Instantiation of the architecture

The presented requirements and guidelines were the main guide for the conception of an actual system capable of dealing with real life access control, which will be presented and explained in this section.

3.3.1 Modules

Having in mind a modular design architecture and relating it with the goals set and the requirements collected, we propose the architecture represented in Figure 3.1. The actores in the architecture are:

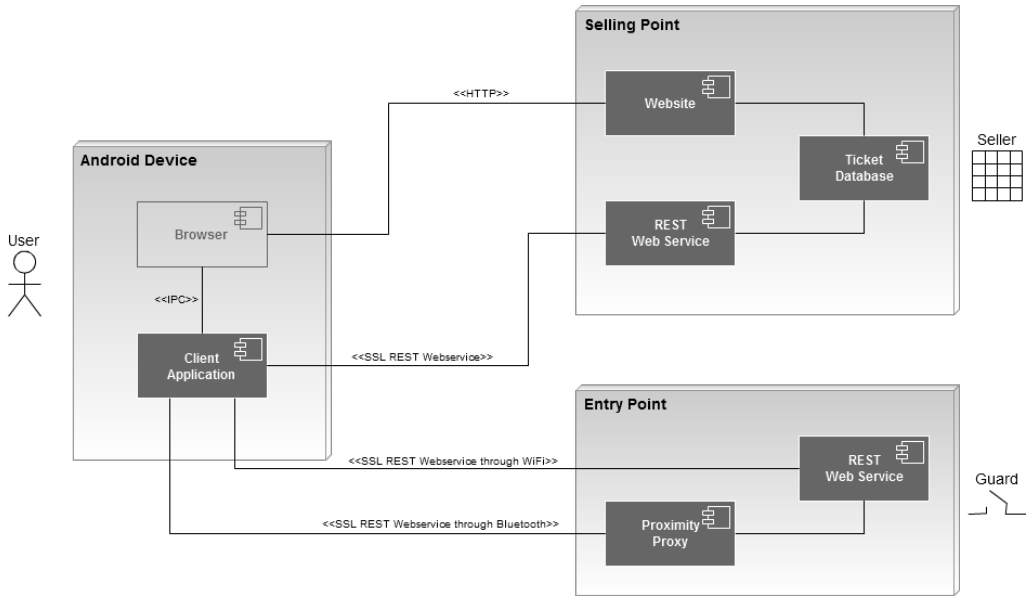


Figure 3.1: Module diagram of the prototype instantiation.

The whole environment of the prototype developed is divided in three main nodes, intrinsically related to their function. Some of the system's modules are shared between them.

Personal mobile device With the correct application, enables its user to buy tickets, checking and managing them, and turning them in later in order to access a restricted location or event. It is responsible for a safe transport mechanism between the two main points: the Selling Point and the Entry Point.

Selling Point It is responsible to issue valid tickets that allow the access to a restricted place or event. It controls the ticket issuing and selling.

Entry Point Module responsible for interacting with the user's mobile device to validate the presented tickets, granting access to those who possess valid credentials.

Each of these nodes entities are formed with modular components as seen in Figure 3.1, each one responsible for an area of the system and independent of each other. These modules and their relations are identified and explained in the following subsections.

3.3.2 Client application

To capacitate a mobile device, like a smartphone, with access control features, a mobile application was developed. This application enables a user's mobile device to initiate a series of actions in order to interact with the web platform, where he can view and buy several tickets to attend events. The application also allows the user to turn in and validate the ticket bought online, in order to certify that his ticket is indeed valid to the intended event. The application was developed to interact with web browsers as part of the usage process.

This application enables the user to buy new tickets, turn in the acquired tickets, and check the previously purchased tickets (ticket information like the artist, the time, the show's geolocation, or even share info from the ticket in social networks), just using his mobile device. It implements the following main features:

- Manages the ticket storage in the device.
- Implements the communication interfaces to the ticket purchase and delivery.
- Discovers Bluetooth consuming services (based on QR Codes and Bluetooth SDP, in our prototype).

During a ticket purchase, the application asks the server for all the data concerning the chosen show and its corresponding ticket. The necessary certificates to the proper ticket validation and resulting entry to the show are some of the necessary information which the application stores for later use. It is noteworthy the fact that this module, the client application, launches the default **web browser** defined in the mobile system for the user to access the online ticket store he desire. This simplifies user interactions.

Upon delivery, the application gives the owner the choice to deliver the ticket. If the user approves it, a secure connection with mutual authentication is established between the two. The information on the ticket the client chose is sent to the server through this connection, in order to verify its authenticity. After proof of its authenticity, the ticket is consumed and the user is granted access to the show.

3.3.3 Selling Point

The service platform consists of a web application whose purpose is to support the purchase and consumption of tickets by the client application. This platform with which users interact using their mobile devices is divided in two main nodes: Selling Point and Entry Point.

Internally, the Selling Point uses the following sub-components:

Selling website: A web application responsible for being the front end of the selling system. It deals with the online ticket viewing and selling. This component carries out the purchase process triggered by the user, which leads to the former ticket generation, so the user could download it.

Ticket database: All of the available tickets displayed in the selling website are stored in this database.

Representational State Transfer (REST) Web service: This is the main back end module, and it is responsible for generating the newly purchased tickets when acting in behalf os the Selling Point. The client establishes a SSL connections with this component every time he buys a ticket.

3.3.4 Entry Point

In turn, the Entry Point is responsible for controlling the access to certain locations, accepting tickets from users in order to assess their authenticity. If proven valid, the user is granted access. This node uses the following sub-components:

REST Web service: It is responsible for consuming the tickets submitted by the users upon entering the show. The client establishes a SSL connections with this component every time he turns in a ticket.

Proximity Proxy: In order to meet the proximity requirement and communicating with the users' devices, the system must possess a Bluetooth endpoint. It is responsible for the communication between Bluetooth Radio Frequency Communication (RFCOMM) and TCP transport channels. This subcomponent of the service provider accepts connections established over Bluetooth and redirects them to a TCP port.

3.3.5 Main Interactions

The interactions with the system are performed relying on the properties of REST web services through a TLS connection. This tunnel, by itself, works as an implicit verification mechanism, since the necessary keys to establish it are present in the ticket.

Below are described the two main system interfaces, involving a user's personal device and the service platform: the first one describes the process of buying/issuing a ticket; the second one reports the process of turning in/consuming a ticket.

Buying a ticket

A ticket purchase between a personal mobile device and the service platform is defined by a set of operations detailed in Figure 3.2:

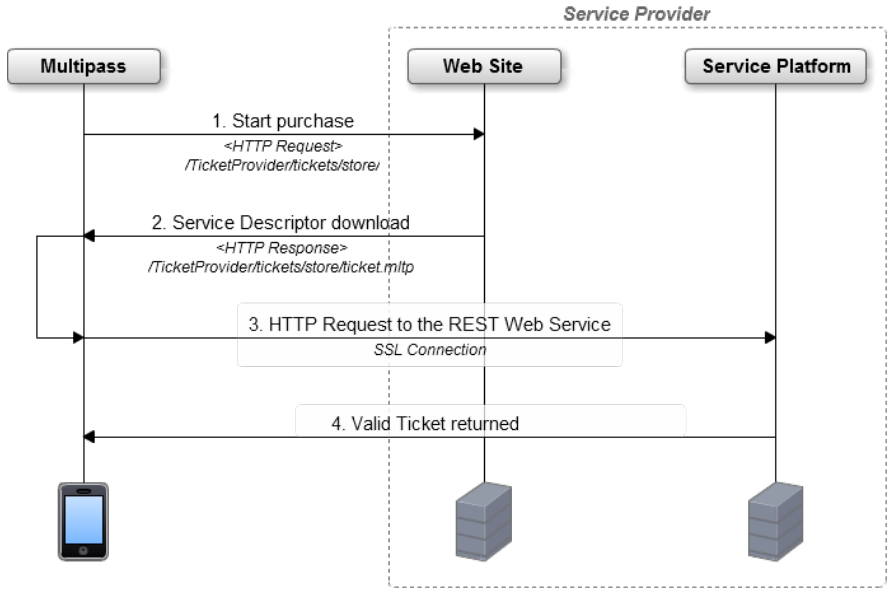


Figure 3.2: Sequence diagram of ticket buying.

The buying process consists of the following steps:

1. The user starts the process in his personal device, using a regular web browser or starting the application, in which he accesses some online store and chooses the ticket he wishes to buy.
2. The selling point generates a XML description of the web service the user's device must reach to generate the ticket, and returns that description as response to the HTTP Request (this service descriptor will be explained later in chapter 4). This action results in a download in the mobile device, where the user triggers the activation of the correct application by clicking in the downloaded file.
3. The application generates a key pair (a public and correspondent private key) and connects with the web service in the received description, sending it the previously generated public key.
4. If the request is valid, the service platform answers with a valid ticket, which the application stores in the personal device, along with the generated private key for the ticket.

Turning in a ticket

This interaction is responsible for consuming the presented tickets by users through their personal mobile devices. The consuming process occurs according to the following diagram:

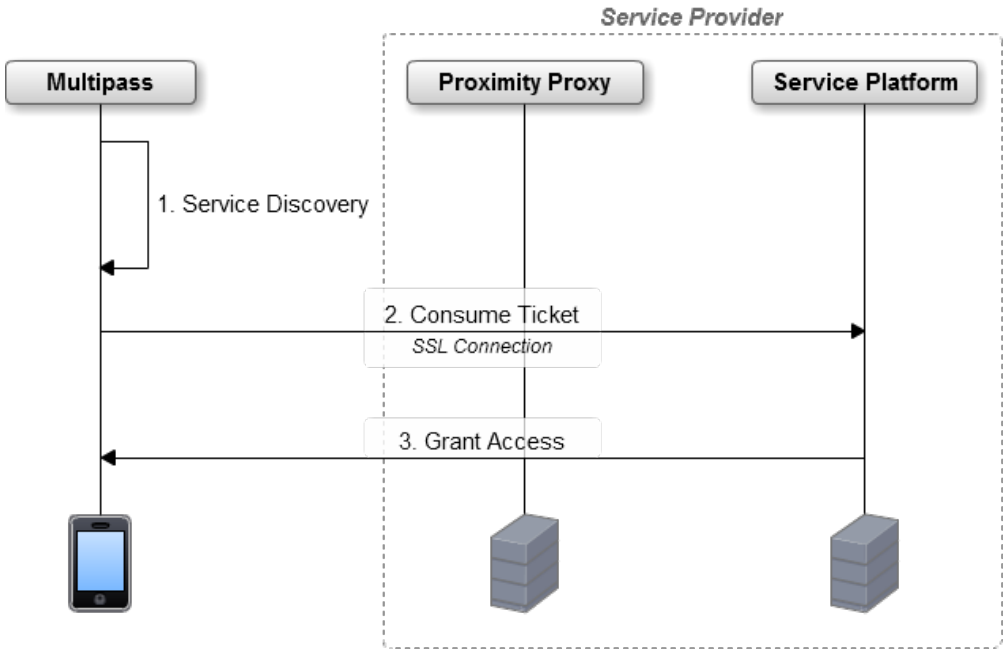


Figure 3.3: Sequence diagram of ticket consuming.

1. The user starts a service discovery using his personal mobile device (through Bluetooth scan or using QR Codes, in our prototype). The result of the discovery is a URL to a TCP/IP or a RFCOMM/Bluetooth channel.

2. Using the above URL, the user's device establishes a connection to the consumer web service in the entry point, and sends his ticket.

3. After receiving the ticket, the entry point verifies its validity by comparing the key used in the TLS tunnel establishment with the key (B) in the ticket and verifying if the ticket was indeed issued by a valid vending point. If the result of these verifications is valid, the entry point grants the requested access to the user.

The connection to the consuming web service can be established in our prototype either by TCP/IP or RFCOMM/Bluetooth. In the event of a Bluetooth connection, it is established through a proximity proxy (Figure 3.3). This additional component is essential because the application server is not capable of handling Bluetooth communications.

In the first step in our prototype, to proceed with the discovery of valid consuming services, a user's device resorts to QR Codes or Bluetooth scanning. The entry point service discovery is better explained in the realisation chapter, in section 4.2.

Chapter 4

Client Implementation

Since most of the work was focused on the client side and its relationship with his portable device, most of this chapter is dedicated to the client application. This is required to explain in detail all the work developed to create a usable, yet secure, client application to be deployed in an Android smartphone. A small section at the end discusses the implementation of the service delivery platform.

The proper realisation of the client application was beyond simple implementation, since the mobile environment deeply influenced the whole system and its settings. The settings, design, and implementation decisions concerning the client application are presented and explained in this section, as well as the application structure and workflow.

4.1 Platform Choice

Android was the chosen mobile platform to develop our application. The reasons of this choice were related to is popularity and easy development. The current main mobile platform are Windows, iOS and Android (presented in no particular order).

Microsoft is present in the mobile market with Windows Mobile, a platform several years old, which is currently being replaced by Windows Phone 7. However, since Windows Phone 7 devices are still reaching the market, their market share and availability are quite small. Its lack of current popularity does not make it an attractive OS to develop the prototype's client application.

Apple uses iOS in their portable devices (iPod, iPhone, iPad) and although it is a mature development platform, it requires Apple hardware to work with it, their development tools are not multi-platform. An Apple computer is required to develop to iOS, which meets with the "walled garden" policy of Apple. Moreover, currently Apple does not provide an official Bluetooth API in its iOS SDK, mostly because of security concerns.

Android is an operating system developed by Google, and currently has the largest share of the smartphone market[68][69][70]. Together with the accessible price of several Android devices, the free and multi-platform SDK are the main reasons why the application was developed in Android.

4.1.1 Development Environment

The Android device used to application deployment and tests was the HTC Wildfire, featuring the 2.2 version of the platform (codename "Froyo", API 8 in the Android SDK). As requested by the proximity requirement, it featured Bluetooth (Class 2) and Wi-Fi capabilities.

The client application was basically developed using the Eclipse IDE (version 6), with the Android plug-in installed, and the Android SDK, properly integrated in the IDE.

To correctly handle tickets and information storage, the client application had two main dependencies: *Bouncy Castle Crypto API*[1] and *Protocol Buffers*[2], both for Java. Bouncy Castle was used to handle the storage of keys and certificates, since Android does not support Java Key Stores (JKSs), it only accepts Bouncy Castle Keystores (BKSs). Protocol Buffers are a more robust and reliable option for serialising objects than native Java solutions, so it used to handle message serialisation in the whole system.

4.1.2 Activity Flow

Taking into account the mobile environment to which the application will be deployed, a clear activity flow must be taken to successfully conceive it.

In Android operating systems, the typical workflow of an application is based on activities. An activity is described as "a single, focused thing that the user can do"[3]. So, the usual flow within an Android mobile application is going from activity to activity, back and/or forth. The prototype application was designed having this flow in mind (Figure 4.1).

4.1.3 Communication

The main focus of communications was the proximity between the two communicating peers, bringing the proximity requirement to the interactions. This way, the client application is able to communicate with the server using Bluetooth (preferably) and Wi-Fi. After the client manifested his intention of turning in a ticket to access a certain show, the application tries to establish a SSL tunnel between the user device and the server, first over Bluetooth and only through Wi-Fi if the previous is not available. Firstly, the encryption provided by this protocol ensures confidentiality of the message content exchanged between the two parties (client and server). Secondly, establishing a SSL connection with the server, based on the server credentials, offers the guarantee the user is connecting to the correct server (assuming its credentials are safe and secret, only known by it); in other words, since the client already knows the entry point public key (provided in the ticket), using it to establish the SSL connection will require the server to possess the same credentials, hence authenticating the server. Thirdly, if client authentication is required by the server, this will also authenticate the client, forcing him to present valid credentials in order to establish the connection.

The nature of the connection is dependent of the service URL patent in the QR Code scanned to access the selected service. This QR Code will have at least on service URL, which

[1] `http://www.bouncycastle.org/java.html`
[2] `https://code.google.com/p/protobuf/`
[3] `http://developer.android.com/reference/android/app/Activity.html`

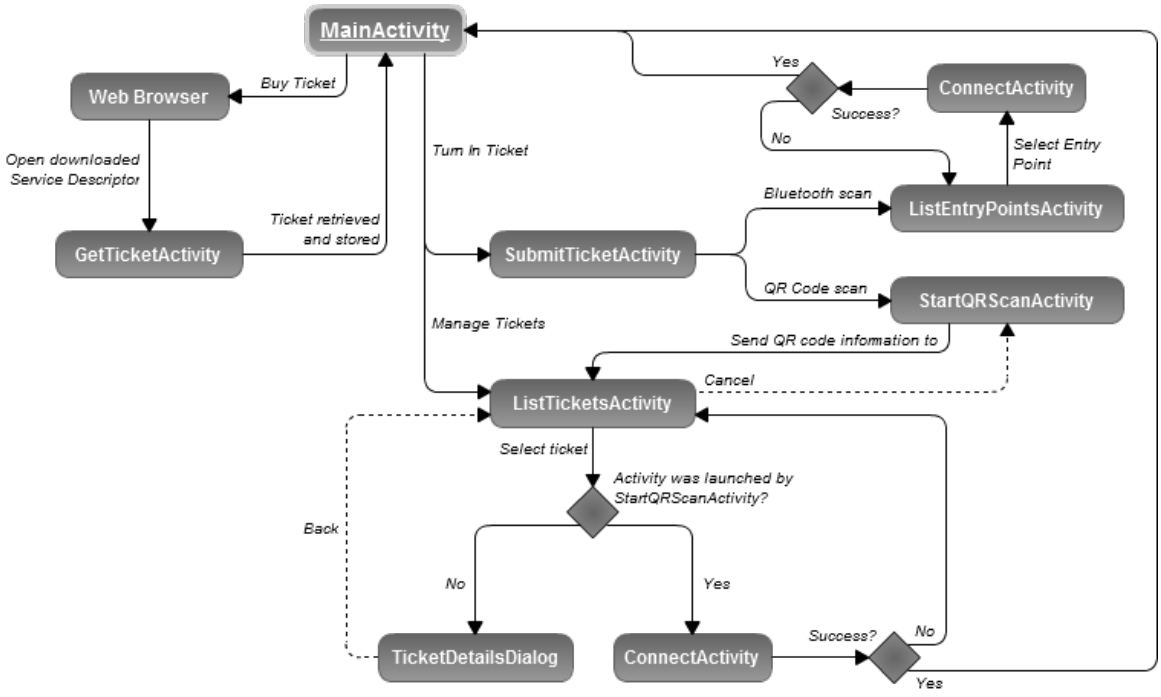


Figure 4.1: Flow diagram of the application activities.

can be a Bluetooth URL (e.g. *rfcomm://*) of an IP URL (e.g. *https://*). To promote closer proximity, Bluetooth URLs are preferable.

4.2 Service Discovery

In order to effectively use the purchased tickets, the user needs to find the corresponding services. A possible problem of Bluetooth Service Discovery is impersonation, i. e., an ill intended person setting a device in the surroundings of a certain service to pose as the correct service point. This is usually performed to trick the users in order to steal their information or something worse. If performed correctly, it is impossible for a user to distinguish between the trustworthy service and the malicious one, making him an easy target. This dissertation introduces the proximity concept as a way to prevent this from happening, but sometimes is not enough to prevent such attacks, since it is possible to simulate proximity while being far away from the authentication spot using an antenna. To prevent service impersonation and proximity simulation, signed QR Codes are introduced in this dissertation as an essential part of the authentication and verification of service discovery.

4.2.1 Signed QR Codes

A QR Code is a type of 2D barcodes (two-dimensional code or matrix barcode), consisting of black modules arranged within a square frame in a (usually) white background. This kind of

barcode is readable by dedicated QR barcode scanners and camera smartphones, and it usually encodes text.

As previously mentioned, the inclusion of QR Codes in the prototype was not a goal, but rather a necessity. This is why this technology was not thoroughly studied and presented in chapter 2, its use was necessary to overcome Bluetooth SDP's limitations in securing service advertising. Yet its use is consistent with the proximity concept, since it forcefully narrows down the interaction scope. The size of the barcode forces the user to be very close to it in order to successfully collect its information, having to place his smartphone just a few centimeters from the QR Code.

QR Codes were used to provide service informations to the users (service advertise), while performing mutual authentication and preventing any tampering attempts on the information contained in the QR Code. The respective fields and info stored in the barcodes basically turn them into signed QR Codes.

This kind of QR Codes contain information which certifies its authenticity, along with equally important information to use the selected service. The QR Code structure obeys the following:

(a) Signed QR Code (b) Unsigned QR Code

Figure 4.2: QR Code examples.

Service URL The first field is mandatory, and indicates the available service point with which the users interact. At least one URL must be presented to the user in the QR Code, although more than one URL may be included. The provider service is transport agnostic, as previously mentioned, and several transport channels may be available. In the implemented prototype for this dissertation, the developed services are available through IP (Wi-Fi) and RFCOMM channels (Bluetooth), and the QR Codes used for the service discovery reflect such capacity, containing 2 URLs. For legacy purposes, only this field is mandatory to accessing a service.

E_P Hash To display only the available tickets for the correct time and place the user is, a hash of the entry point public key is available in the QR Code. The user mobile device compares this hash with a hash of the entry keys it has from the purchased tickets, and displays the tickets it found a match.

E_K Signature To frustrate any QR Code forging attempt, like copying the information available in a QR Code replacing only the service URL to perform an attack, the final field of the information represented in the signed QR Code is a signature of the previous information using the entry point private key. This signature can be verified using the entry public key, present in the user device (by storing it during the ticket purchase).

Figure 4.2 presents two similar QR Codes regarding the fields they encode. The difference between 4.2(a) and 4.2(b) is the last field, the E_K Signature. Both QR Codes have the Service URL and E_P Hash, but 4.2(a) also encodes a signature of the previous two fields using E_K. This way, the information encoded in a signed QR Code is assuredly valid, and the client application is able to filter any barcode without a valid signature by an entity it knows.

4.2(a) is a signed QR Code because its Service URL and E_P Hash were signed with E_K,

4.2.2 Bluetooth SDP

By taking advantage of Bluetooth SDP, it is possible to have entry points advertising special services for ticket delivery. The client application, when searching for available entry points to turn in a ticket, would only display to the user the tickets he can turn in considering the active entry points in that location and time.

The Android Bluetooth API is maturing from one version to another, but it is still not evolved enough to give developers access to service scanning and publishing. Thus, it is not possible to access service capabilities, like searching for a certain ticket delivery service, even if it is available. This way, the built prototype does not focus on this method of discovery, but rather on the QR Codes method.

4.3 Ticket Settings

Since the prototype revolves around ticket handling, it makes sense to describe and explain the main decisions behind the ticket environment and handling. We decided to present those settings in the client implementation section because this system aspect is mainly centred around the user and his use of the tickets. Most of the decisions where taken based on the mobile environment where the ticket would be stored, the communication mean (wireless), and Android software characteristics.

Service Descriptor

As described in chapter 3, when a user buys a ticket to a show through the online store, a description file is downloaded to the user's device, a file with the *application/x-multipass-wsd* MIME type and *mltp* file extension. This is basically a XML file with the following structure:

```
<ticket>
  <ws>http://foo.bar.com:8080/TicketProvider/tickets/retrieve/</ws>
  <token>78549839485487</token>
</ticket>
```

This file includes the two pieces of information required to retrieve the purchased ticket: the web service where to download the real ticket, and the info token to present which addresses the ticket to be retrieved. From a user's perspective, all he has to do is 'open' this description file: the device will automatically handle the file opening of *mltp* files with the application of the prototype, which in turn retrieves from the file descriptor the necessary info to request the ticket.

This intermediate XML file was introduced specially due to some Android limitations. It was not possible to intercept a user's click in a link in the web browser in order to pass the service arguments to the client application, so this service descriptor file was introduced. After downloading it, the user must open the file by clicking in it. Again, Android limitations forced the inclusion of this step, since the OS does not allow the setting of a default application to automatically open a downloaded file. The user must actively open the recently downloaded file itself.

Although forced, the use of the service descriptor has two benefits:

1. Allows the client to interact with online ticket stores using only a browser in his device, without using any special extension.
2. Minimises the necessary changes for adapting the selling ticket website to support the e-Ticketing system of the prototype.

Ticket Format

An issued ticket is structured according to Figure 4.3:

Figure 4.3: Ticket structure, after its issue.

The information fields of a ticket are the following:

B_P Ticket public key, generated by the user's device. The corresponding private key is B_K, which is stored in the device.
Metadata Information regarding the ticket purpose, like the name of the show or GPS coordinates.
E Entry point certificate.
V Vendor certificate.
Sig(V_K) Signature performed with the private key from the vendor's certificate.

This structure ensures the ticket validity by its signature with the private key associated with the certificate (V), the authentication entry point by the certificate (E), and the device authentication by the B_P.

Ticket Credentials

A main element for ticket validation is its own key pair, its public and private key. This key pair is generated upon ticket creation, on the user's personal device, since the ticket private key is supposed to be secret, hence never leaving the user's device.

The application generates a key pair (RSA keys of 2048 bits each, to provide extra security[71]), which will be the keys (public and private) of the ticket. Both will be saved in a key store on the user device, using a master password. This public key will be shared with the server at the ticket request, using the info in the service descriptor. The application does not use 4096 bytes RSA keys because the limited computational power of the mobile devices has to be considered, and 2048 bytes were deemed to be "enough security".

If this request is successful, the server will send the requested ticket through the established SSL tunnel, according with the structure described earlier. Upon reception, the ticket is saved in the folder where the application previously saved the ticket credentials, partitioned as the following:

- A file stores all the information received, considering it as the whole ticket.
- A key store saves the Entry Point certificate separately, for easier later access.

4.4 User Interface

Since the developed application aims to be easily used by people, its user interface and general usability are important issues to address to. This way, the user interface and application flow is intended to be simple and clear from a user's perspective. The home screen displays the Multipass project logo and the main actions to perform.

Pressing 'Buy ticket' (Figure 4.4) opens a web browser, allowing the user to access any online ticket store which supports this e-Ticketing system (Figure 4.5).

Clicking in 'Submit ticket' will display the service discovery activity. In here, the user is able to search for services to which he is able to turn in previously bought tickets, whether by Bluetooth scanning or QR Code recognition. Once he finds a suitable service, he can turn in the ticket to finally gain access to the desired location or show.

'My tickets' will display a list of all the purchased tickets still not used (Figure 4.7). The user will then be able to access and check details of the tickets he possesses, and even share information regarding his tickets in social networks.

Before any important action can be performed, the application demands user's feedback (Figure 4.8). He is asked for confirmation about the operation, because all of the interactions involve information exchange and possibly the change of some condition (e.g., unconsumed ticket to consumed ticket, hence rendering it useless for later use). These operations must be always confirmed by the user in order to raise his awareness for the current action being performed and its consequences.

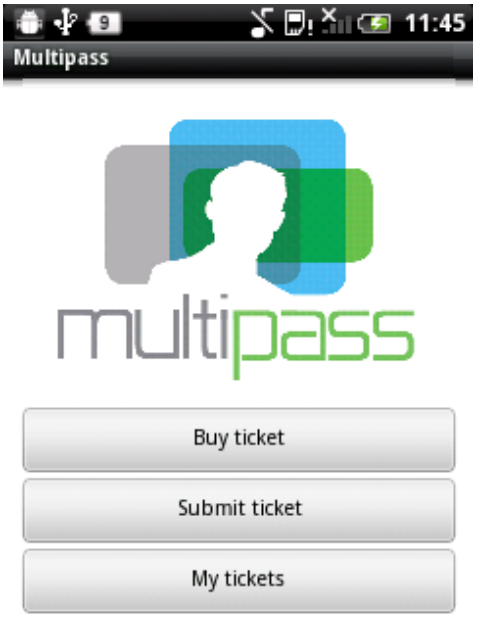


Figure 4.4: Home screen of the application.

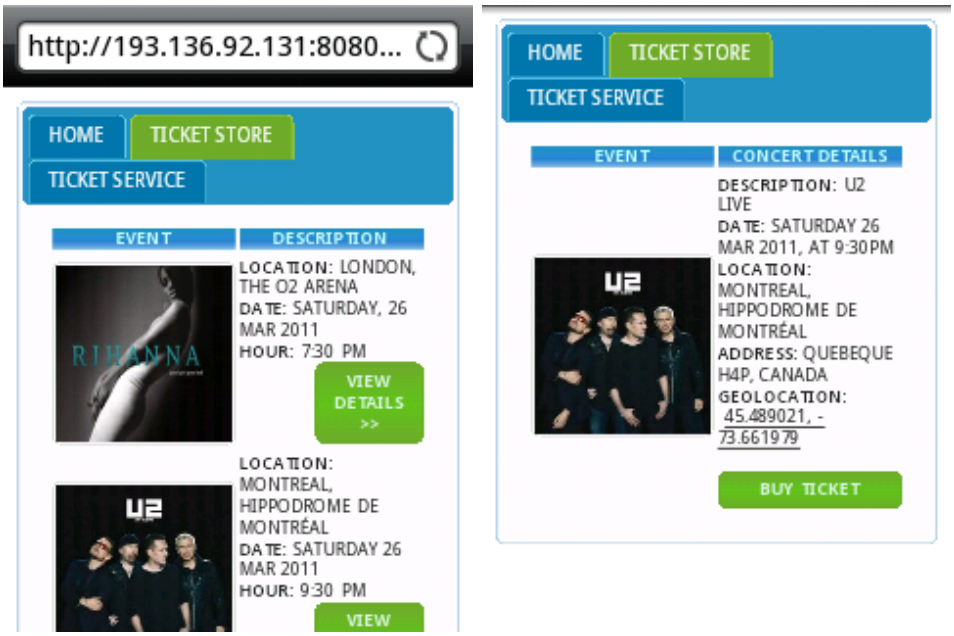


Figure 4.5: Example screen of "Buy ticket".

4.5 Functional Evaluation

After the system presentation and the application description, a comment about the end result is necessary for a more cohesive comprehension about the prototype of this dissertation. The evaluation of the whole application will be performed regarding the key requirements and

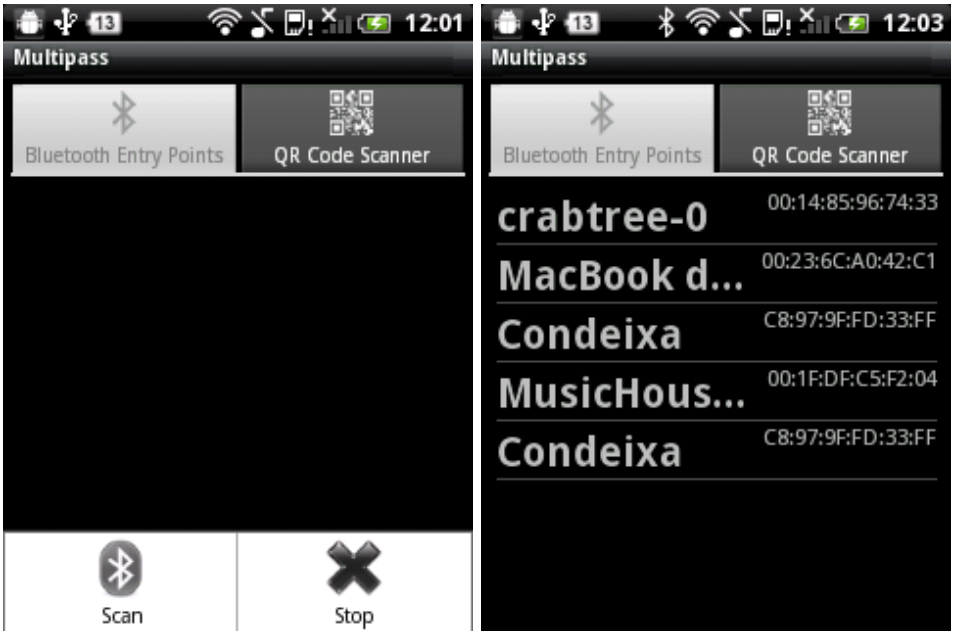


Figure 4.6: Example screen of "Submit ticket".

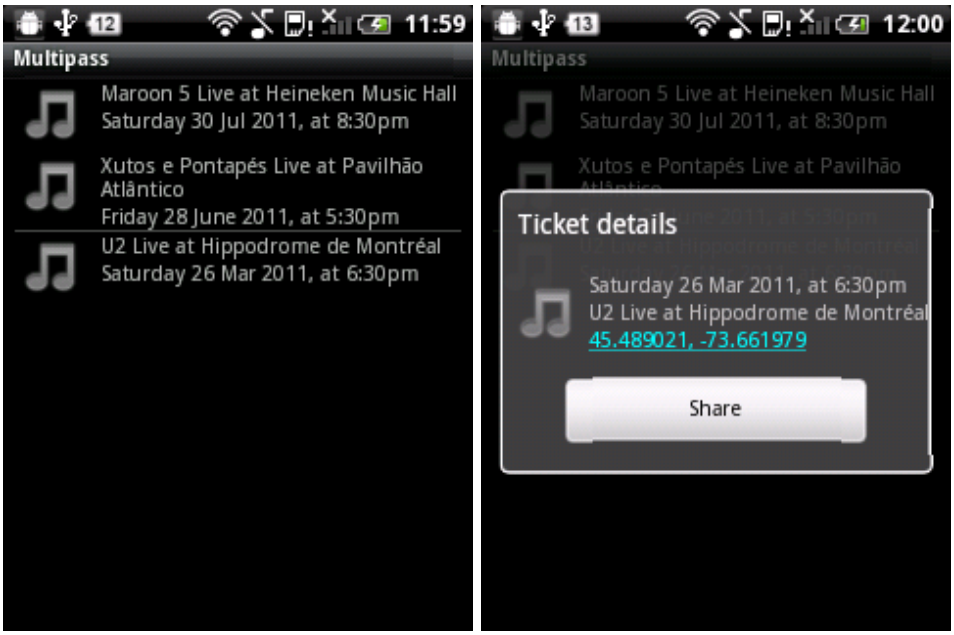


Figure 4.7: Example screen of "My tickets".

guidelines established.

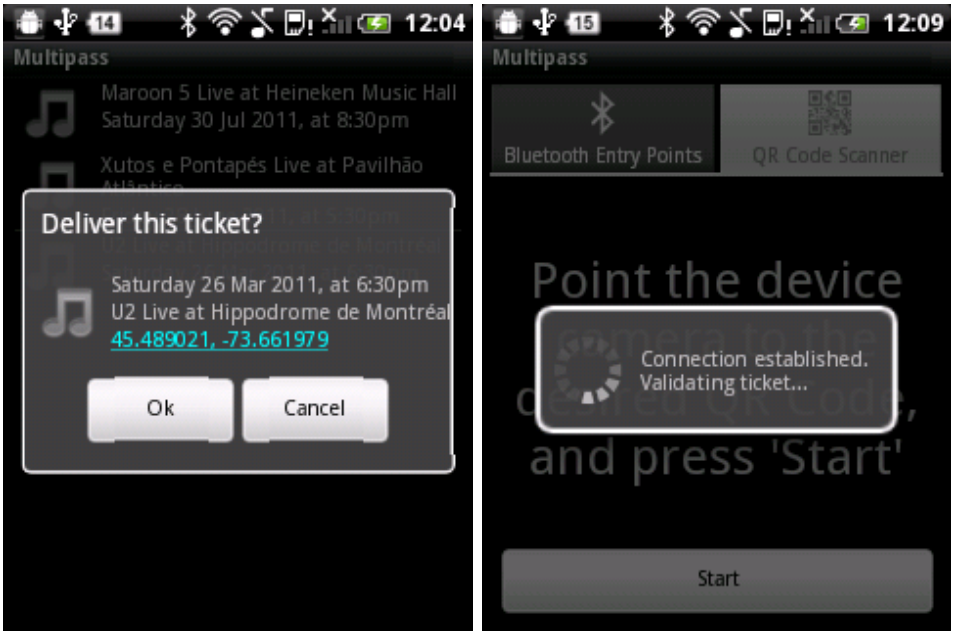


Figure 4.8: Application asking for user confirmation.

4.5.1 Usability

The special emphasis given to the client application makes its usability a characteristic to be considered and evaluated. To assess aspects of the application, a simple usability test was performed. 15 users trialled the application, representing an audience within the 21 to 27 years old range, constituted of 10 male users and 5 female users. The target audience was selected in order to represent an age group which uses smartphones in their daily lives, and are early adopters of technology. For comparison purposes, a user familiar with our client application was also tested (male, 23 years old).

Three basic tasks of the e-Ticketing scenario were evaluated:

Task 1: Buy a ticket
Task 2: Check details of the ticket purchased
Task 3: Turn in the purchased ticket

In Figure 4.9, it is possible to see the total time of the three tasks performed by the 15 subjects, along with the baseline subject. As can be seen, task 2 is the shortest one, and represented no trouble for any user. In general, task 1 was the longest one, ahead of task 3. Our observations also shown that task 1 was the one with more mistakes by the users. Through a simple enquiry every user answered after the experiment, it was possible to determine some conclusions:

- Most of the users had experience in using smartphones.
- The least experienced users in using smartphones were the ones who took the longest to perform the tasks.

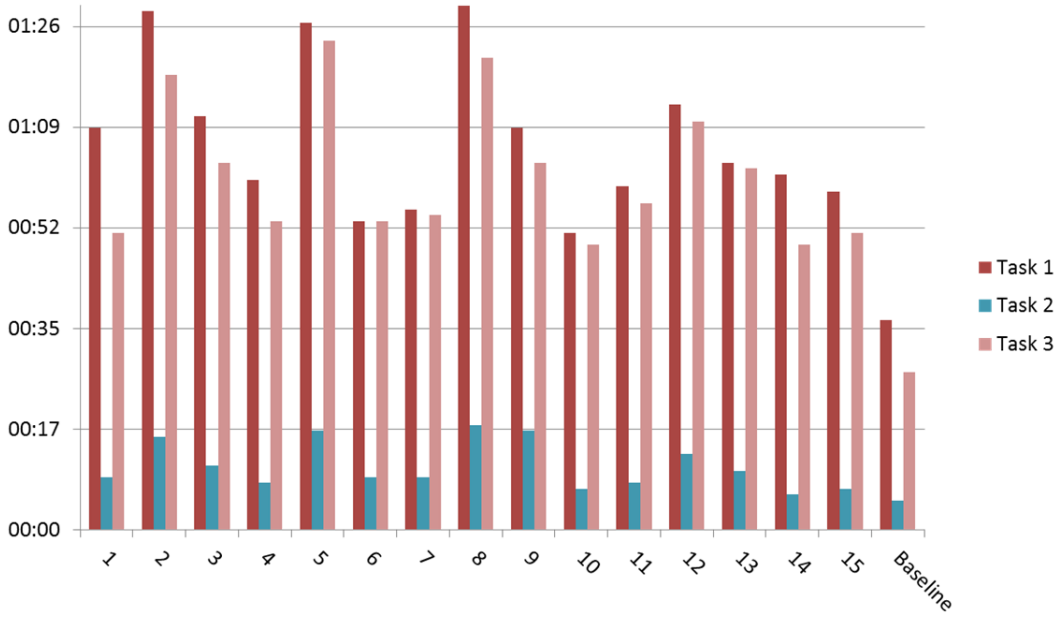


Figure 4.9: Execution times of the three usability tasks evaluated.

- Apart from task 1, almost no mistakes were made using the application.
- The hardest part for all of the subjects while using the application was understanding of what to do with the service descriptor. The users who had most difficulty figuring out that they were expected to click in the service descriptor downloaded to continue the buying process were the ones with no experience using smartphones.
- 80% of the users found useful an application that allows them to buy, manage, and turn in tickets.
- Most of the users (87%) also found useful to extend the features of the application to integrate credit and ID cards, for example.
- The majority of the users (73%) answered that they trusted in this kind of applications, even considering the sensitivity of the information it handles.

4.5.2 Proximity

Since proximity is the cornerstone of the prototype presented and the main focus of this dissertation, it could not be left aside this evaluation.

As recalled, Bluetooth was the short-range wireless technology chosen to perceive the physical presence between the user's mobile device and the access control system. The personal mobile device used in the project, a HTC Wildfire, featured a Class 2 Bluetooth adapter, which is able to establish Bluetooth connection up to about 10 meters. When the proximity concept was introduced, it was referred that the smaller the area is, the easier it is for the user to have some kind of control over it: the smaller the area, the fewer threats can exist. It seems a rather simplistic view, but it basically ends up working this way when it comes to proximity.

So, considering the range of the mobile device, it is plausible to say that Bluetooth might not be the best short-range wireless technology to be used in proximity-based systems. NFC could become a viable alternative in a near future, since new and feature-rich NFC powered mobile devices are coming to the market. The NFC wireless range of these devices will be only of a few centimetres, which enhances the sense of proximity and restricts the possibilities of attack, as previously mentioned.

Bluetooth Communications

Bluetooth was the technology applied to turn in tickets to entry points, as already mentioned. From this fact comes certain properties that are inherent in this technology.

The tests made to the prototype were performed in an environment rich in wireless communications in the ISM band, in which Bluetooth also operates. This means high chances of collision between wireless communications, and eventually affects Bluetooth performance. However, this scenario was chosen since it simulates current urban environments, were Wi-Fi is pervasive.

In the tests, Bluetooth connections were fairly reliable (rarely occuring any error while establishing connections), however they took longer than desired for an e-Ticketing solution (entry monitoring actions should take as short as possible in order to maximise the number of people getting in, without neglecting the proper access control). According to the Bluetooth specification, device inquiry should last for more than 10 seconds[72][73] unless the inquirer collects enough responses for its purpose and thus resolves to abort the inquiry earlier. Connection establishment is also time consuming, taking around 5 seconds in a typical case and as long as 23 seconds in the worst case[73]. The Bluetooth protocol stack implementation on Android is not meant to rapidly establish connections between devices, and adding all of the environment radio interference, the act of turning a ticket took around half a minute for an experienced user. The best time value for turning in a ticket would be just a few seconds, and Bluetooth does not seem fit for such task. So, at least in this setting, perhaps the biggest drawback of Bluetooth is its latency. The establishment of a Bluetooth connection between two devices it is far from being instantaneous, even if the devices are already paired with each other. The very scan of devices takes some seconds, making it impossible to have almost instant Bluetooth connections. In this case, the use of NFC would probably speed up the connection setup and even nullify the need for device pairing, like Bluetooth requires.

The prototype realisation had its hardware and software limitations. Currently, most Bluetooth enabled devices feature version 2.0 or 2.1 of Bluetooth stack. This means most devices can not take advantage of advanced features like transfer speeds of 24 Mbit/s, enhanced power control, or data transmission without explicitly establish L2CAP channels. In addition, the stacks Bluetooth devices provide are not sufficient to use their features. The OS must provide interfaces to access and use such capabilites. Unfortunately, Android Bluetooth API is relatively recent (introduced in 2009 in Android 2.0 “Eclair”), so its supported features are few and limited. For example, there is no way to perform service discovery. This feature would be very useful in the prototype developed for this dissertation, since it would be possible to set the client application to list only Bluetooth devices with a custom service that only Entry Points could advertise. Android also does not provide access to Bluetooth security features and modes, encrypting the communications by default if the connected devices support it.

4.5.3 Security

Two areas of security application are important in this dissertation: communicating, which concerns the secure transmission of information between devices, and storing, which refers to the way the generated and received data is safely stored in the client-side.

In addition to the security measures that the wireless technology already applies, in the prototype we were able to establish SSL connections over Bluetooth when turning in a ticket. This connection is established using the entry point's certificate as credentials, which came with the ticket. If mutual authentication is required, the client has to provide valid credentials as well, using a certificate with the credentials generated for the ticket he intends to turn in.

With the use of SSL, the security of the connection is virtually assured. Most of the attacks to SSL are linked to the implementation instead of the protocol itself. The latest version of TLS is one of the best ways to ensure private communications over a public network, and it is common to consider safe any communication performed through a connection secured by this protocol.

Regarding the ticket storage, the only security measure implemented for the prototype was storing all the credentials in password protected keystores, since safe communication between devices was the privileged security feature. However, several mechanisms could be implemented in order to design a very safe storage environment. For example, when tickets are received by the client, they could be dismantled in their respective fields and stored in a database, encrypting the data before storing it and protecting the database with a password.

Mutual Authentication

The use of TLS in this dissertation allowed to take advantage of its client authentication feature. When enabled, this requirement forces the client to provide credentials upon connection establishment. The server is responsible for allowing or not these credentials, depending on the level of requirement set.

In the dissertation's prototype, mutual authentication is performed in two ways. First, when a user approaches an entry point and scans the entry point's QR Code to discover the correct service to turn in his ticket, the information contained in the QR Code provides the client application with enough data for it to assess if the scanned QR Code presents valid (and familiar) credentials, thus enabling the application to verify if it possesses any valid ticket for that entry point. Secondly, with the Mutual Authentication enabled in TLS connections, the access control system requires the client to present valid credentials. The client application must use the ticket's credentials to successfully establish a connection with the server. These two measures ensure the access control service authenticity (by forcing it to have the correct credentials in the QR Code to be scanned in the entry point) and it also ensures the client possesses a valid ticket (by only displaying the respective tickets for the scanned info).

4.5.4 Ticket Handling

The performance of the chosen scenario must be evaluated too, since it may provide us information about the possible difficulties in implementing a system of its kind in the near

future. Therefore, some of the main aspects of the e-Ticketing prototype were evaluated and assessed, in order to get an idea of the overall system performance.

Error Handling

As any system, this environment is prone to errors. Events such as losing connectivity or malformed received data are real hazards in any system, the prototype developed is no exception. To minimise or even nullify unpleasant consequences resulting from these events, actions such as render useless a ticket after its use, for example, must be executed after proper confirmation of the system.

Disconnections between the client and the server just interrupt the natural course of the buying or turning in process, forcing the user to repeat the process. If the process of buying a ticket is interrupted, the user just has to access the ticket store again (if the download of the service descriptor was unsuccessful) or open the service descriptor downloaded again (if the service descriptor was successfully downloaded and the process was interrupted when the client application was trying to retrieve the ticket). If the interrupted process was the ticket delivery to the entry point, the user just has to initiate it again.

Consumed Tickets

Once a ticket has been successfully turned in to an entry point, it is consumed and therefore useless. The used ticket, instead of being deleted since it has no more practical value, should be stored in order for its owner to have a record of the shows (or other events) he went.

The prototype client application flags a ticked as "used" when it is turned in, keeping it alongside the unused tickets. Future work could store the information of the used tickets in a database and delete the actual tickets after their use.

Chapter 5

Conclusions

The primary focus of this dissertation was the study and integration of proximity in secure authentication scenarios. To accomplish such goal, a prototype of a mobile application using proximity communications to turn in digital tokens was designed and implemented, along with an access control system to support the proper use of this application.

The access control system is built upon a service platform which provides ticket selling and consuming services. With the mobile application, it is possible to buy tickets (digital tokens) in the store website, check the details of the bought tickets, and submitting them afterwards using short-range wireless technologies to integrate the proximity concept into the authentication process. The mobile application is the central piece of the system, and it was designed to be a safe, interactive and usable way to store, transport and exchange private information (in this case, digital tokens representing tickets). The application was designed to be modular, specially regarding the wireless technology to be used to establish secure proximity-based connections with access control systems. To ensure communication privacy between the application and the access control system, an SSL tunnel is always established in the proximity-based interactions, being agnostic of the transmission medium. Public-key cryptography elements, such as public and private keys and certificates were used throughout the dissertation to digitally verify and certify the tokens and the entities issuing them. Finally, it is also important to note the usability of the mobile application, which was taken into account in order to implement a secure, yet user-friendly, application capable of allowing a regular person to get, manage, and exchange digital tokens.

The final mobile application met most of the requirements and guidelines presented in chapter 4, and achieved the goals set by using Bluetooth as the preferred proximity mean for authentication, by integrating public-key cryptography elements and methods to assess and attest the authenticity of digital tokens and entities, and by using TLS connections to safely exchange information between peers.

This work aimed to present a prototype capable of providing a modern, safe, and usable proximity-based service without complex solutions or bleeding edge technology. The whole prototype presented was designed having in mind a demonstration of a likely future scenario, where smartphones will be increasingly important in our daily lives. They will be part of everyday actions and just as important as our wallets. People will interact with their surroundings through short-range wireless communications, whether actively (Device to Human interactions) or passively (Device to Device interactions) using their personal mobile devices. More importantly,

proximity will be an ubiquitous concept in these interactions, being inextricably linked with the wireless technologies used. It allows safer communications and requires closeness between the two interacting parties. Our cellphones, smartphones, notebooks and tables are already equipped with these proximity wireless technologies, yet most of their capabilities are not explored fully. Few applications enable and encourage proximity-based interactions between devices, and most of it are just location-based, i.e., the closeness between devices is evaluated using GPS data. With such wireless technologies built into these devices, it is natural to explore their possibilities and to build useful services that take advantage of them.

5.1 Future Work

With the increasing ubiquity of short-range wireless technologies, much more research will be necessary to tackle future problems and correctly integrate such technologies in our daily life with added value for the end user.

For instance, technologies like RFID and NFC must be improved in the security field, since they have no "native" security or privacy mechanism and NFC enabled services are starting to appear, specially for mobile payments. More and more sensitive operations are currently being performed through wireless mediums, and their security is crucial to protect the exchanged information and the participating entities.

Access control systems like the prototype presented in this dissertation have much to gain from integrating Identity Management units into their back-ends. This provides an extra layer of security, privacy, and flexibility, since it would be able to store less sensitive information in the personal mobile devices (digital tokens would be stored in the user's account in the IdM back-end, for example) and identity verification and proper user authentication would be more accurate, since the IdM back-end could keep more information about the user and maintain an interaction report of the user, for example.

Furthermore, the mobile application developed would receive a major improvement in terms of security if a more complex storing mechanism would be implemented. For example, data could be striped from the digital tickets received and stored in a database, or the received ticket could be compressed in order to save disk space, protecting it with a master password.

Finally, the implemented prototype is just a proof-of-concept of simple digital interactions relevant in our daily lives. Therefore, other features like delegating tickets from a user to another, or use NFC to discover entry points' services or rapidly establish connections between devices, are some examples of future possibilities.

Bibliography

[1] R. Anderson, "Can we fix the security economics of federated authentication?" in *SPW 2011, 19th International Workshop on Security Protocols*, ser. Lecture Notes in Computer Science, J. A. Malcolm, Ed. London, UK: Springer-Verlag London, Mar. 2011. [Online]. Available: http://spw.stca.herts.ac.uk/2.pdf

[2] B. Schneier, "Balancing security and usability in authentication," 2009, [Online; accessed 12-June-2011]. [Online]. Available: http://www.schneier.com/blog/archives/2009/02/balancing_secur.html

[3] G. Rose, Q. Li, and L. Xiao, "Multiple fingers synchronization used for device mutual authentication," in *Biometrics, Identity and Security (BIdS), 2009 International Conference on*, 2009, pp. 1 –8.

[4] C. Decker, S. Nguissi, J. Haller, and R. Kilian-Kehr, "Proximity as a security property in a mobile enterprise application context," in *Proceedings of the Proceedings of the 37th Annual Hawaii International Conference on System Sciences (HICSS'04) - Track 7 - Volume 7*, ser. HICSS '04. Washington, DC, USA: IEEE Computer Society, 2004, pp. 70 189.2–. [Online]. Available: http://portal.acm.org/citation.cfm?id=962755.963135

[5] A. Mishra, S. Rayanchu, A. Shukla, and S. Banerjee, "Towards secure localization using wireless 'congruity'," in *Proceedings of the Eighth IEEE Workshop on Mobile Computing Systems and Applications*, ser. HOTMOBILE '07. Washington, DC, USA: IEEE Computer Society, 2007, pp. 3–8. [Online]. Available: http://dx.doi.org/10.1109/HOTMOBILE.2007.18

[6] W. Jansen and V. Korolev, "A location-based mechanism for mobile device security," in *Proceedings of the 2009 WRI World Congress on Computer Science and Information Engineering - Volume 01*, ser. CSIE '09. Washington, DC, USA: IEEE Computer Society, 2009, pp. 99–104. [Online]. Available: http://dx.doi.org/10.1109/CSIE.2009.719

[7] "Free your pockets," [Online; accessed 29-June-2011]. [Online]. Available: http://thinkquarterly.co.uk/01-data/free-your-pockets/

[8] "Comming soon: make your phone your wallet," [Online; accessed 29-June-2011]. [Online]. Available: http://googleblog.blogspot.com/2011/05/coming-soon-make-your-phone-your-wallet.html

[9] J. Cheng, "Get discounts, pay with your phone with google wallet, offers," [Online; accessed 29-June-2011]. [Online]. Available: http://arstechnica.com/gadgets/news/2011/05/get-discounts-pay-with-your-phone-with-google-wallet-offers.ars

[10] Y. He, R. Yuan, X. Ma, and J. Li, "The ieee 802.11 power saving mechanism: An experimental study," in *Wireless Communications and Networking Conference, 2008. WCNC 2008. IEEE*, 2008, pp. 1362 –1367.

[11] G. Anastasi, M. Conti, E. Gregori, and A. Passarella, "802.11 power-saving mode for mobile computing in wi-fi hotspots: Limitations, enhancements and open issues," *Wireless Networks*, vol. 14, pp. 745–768, 2008, 10.1007/s11276-006-0010-9. [Online]. Available: http://dx.doi.org/10.1007/s11276-006-0010-9

[12] ——, "A performance study of power-saving polices for wifi hotspots," *Computer Networks*, vol. 45, pp. 295–318, 2004.

[13] "802.11g wireless internet access," [Online; accessed 28-June-2011]. [Online]. Available: http://www.bbwexchange.com/wireless_internet_access/802.11g_wireless_internet_access.asp

[14] D. G. Tarrago, "Home wireless security and privacy: A practical protocol mixing," *Advanced International Conference on Telecommunications*, vol. 0, pp. 7–12, 2010.

[15] M. Hanlon, "World's smallest and lowest power wifi chipset solution," 2006, [Online; accessed 18-June-2011]. [Online]. Available: http://www.gizmag.com/go/5763/

[16] C. Zibreg, "16gb wifi ipad components cost $260, iSuppli estimates," 2010, [Online; accessed 18-June-2011]. [Online]. Available: http://www.geek.com/articles/news/16gb-wifi-ipad-components-cost-260-isuppli-estimates-2010047//

[17] A. Birk, "IrDA lecture," 2003, [Online; accessed 18-June-2011]. [Online]. Available: http://www.faculty.iu-bremen.de/birk/lectures/PC101-2003/17bluetooth/bluetooth/irda.html

[18] D. Suvak, "Extended systems, inc. "IrDA and Bluetooth: A complementary comparison"."

[19] G. Diviney, *An Introduction to Short-Range Wireless Data Communications*, 2003.

[20] S. Williams, "IrDA: Past, present and future," 2000.

[21] M. Nilsson and J. Hallberg, "Positioning with bluetooth, irda, rfid," Master's thesis, Luleå University of Technology, 2002.

[22] D. Axtman, A. Ogus, and J. Reilly, *Infrared data association LAN access extensions for link management protocol, IRLAN*, Jul. 1997.

[23] "Bluetooth on the road," [Online; accessed 17-June-2011]. [Online]. Available: http://www.hoovers.com/business-information/--pageid__13751--/global-hoov-index.xhtml

[24] G. A. Francia, A. Kilaru, L. Phuong, and M. Vashi, "An empirical study of bluetooth performance," in *ACM International Conference Proceeding Series*, 2004.

[25] J. Linskey, "Bluetooth and power consumption: issues and answers," 2001.

[26] L. Negri and L. Thiele, "Power management for bluetooth sensor networks," in *Wireless Sensor Networks*, ser. Lecture Notes in Computer Science, K. Römer, H. Karl, and F. Mattern, Eds. Springer Berlin / Heidelberg, 2006, vol. 3868, pp. 196–211. [Online]. Available: http://dx.doi.org/10.1007/11669463_16

[27] R. Bouhenguel, I. Mahgoub, and M. Ilyas, "Bluetooth security in wearable computing applications," in *High Capacity Optical Networks and Enabling Technologies, 2008. HONET 2008. International Symposium on*. IEEE, 2008, pp. 182–186.

[28] C. Hager and S. Midkiff, "An analysis of bluetooth security vulnerabilities," in *Wireless Communications and Networking, 2003. WCNC 2003. 2003 IEEE*, vol. 3, 2003, pp. 1825 –1831 vol.3.

[29] J. C. Haartsen, "The bluetooth radio system," *IEEE Personal Communications*, 2000.

[30] K. Scarfone and J. Padgette, "Guide to bluetooth security," Tech. Rep., 2008.

[31] J. T. Vainio, "Bluetooth security," 2000. [Online]. Available: http://www.cse.tkk.fi/fi/opinnot/T-110.5190/2000/bluetooth_security/bluesec.html

[32] G. Dini and M. Tiloca, "Considerations on security in zigbee networks," *Sensor Networks, Ubiquitous, and Trustworthy Computing, International Conference on*, vol. 0, pp. 58–65, 2010.

[33] L. Cao, Y. Liu, and S. Yang, "Wireless networked security system based on zigbee technology," in *Wireless Communications, Networking and Mobile Computing, 2008. WiCOM'08. 4th International Conference on*. IEEE, pp. 1–4.

[34] C. Misal, "Analysis of power consumption of an end device in a zigbee mesh network," Ph.D. dissertation, The University of North Carolina, 2007.

[35] Y. Chengbo, L. Yanfei, and L. Li, "Research and application on the coverage range of the zigbee protocol," in *Information, Computing and Telecommunication, 2009. YC-ICT '09. IEEE Youth Conference on*, 2009, pp. 1 –4.

[36] M. Khan, R. Passerone, and D. Macii, "Fzepel: Rf-level power consumption measurement (rf-pm) for zigbee wireless sensor network-towards cross layer optimization," in *Emerging Technologies and Factory Automation, 2008. ETFA 2008. IEEE International Conference on*, 2008, pp. 959 –966.

[37] H. Li, Z. Jia, and X. Xue, "Application and analysis of zigbee security services specification," in *Proceedings of the 2010 Second International Conference on Networks Security, Wireless Communications and Trusted Computing - Volume 02*, ser. NSWCTC '10. Washington, DC, USA: IEEE Computer Society, 2010, pp. 494–497.

[38] "Ember unveils industry's highest performance zigbee chips," [Online; accessed 21-June-2011]. [Online]. Available: https://www.zigbee.org/Knowledgebase/PressReleaseView.aspx?1=1&moduleID=718&Contenttype=ArticleDet&ArticleID=216

[39] "Probee embedded zigbee module-chip antenna," [Online; accessed 21-June-2011]. [Online]. Available: http://www.kanda.com/products/Sena/ZE10C-00.html

[40] "Packet micro - zigbee (ieee 802.15.4) module," [Online; accessed 21-June-2011]. [Online]. Available: http://www.packetmicro.com/ZigBee.html

[41] S. Lahiri, *RFID Sourcebook*. USA: IBM Press, 2006.

[42] A. Juels and S. A. Weis, "Defining strong privacy for rfid," Tech. Rep., 2006.

[43] H. Giess, "Do you want to know more? rfid tags on stationary batteries," in *Telecommunications Energy Conference, 2006. INTELEC'06. 28th Annual International*. IEEE, 2006, pp. 1–6.

[44] I. Yamada, S. Shiotsu, A. Itasaki, S. Inano, K. Yasaki, and M. Takenaka, "Secure active rfid tag system," in *In 4th Workshop on UbiComp Privacy*, 2005.

[45] M. W. Cardullo, "Transponder apparatus and system," Patent US 3 713 148, 05 21, 1970.

[46] H. Cho and Y. Baek, "Design and implementation of an active rfid system platform," *Applications and the Internet Workshops, IEEE/IPSJ International Symposium on*, vol. 0, pp. 80–83, 2006.

[47] G. P. Hancke, "Practical attacks on proximity identification systems (short paper)," in *IEEE Symposium on Security and Privacy*, 2006, pp. 328–333.

[48] D. Dressen, "Considerations for rfid technology selection. atmel applications journal. corporate communication," *Atmel Corporation*, no. 3, 2004.

[49] D. Goodin, "Passport rfids cloned wholesale by $250 ebay auction spree," [Online; accessed 19-June-2011]. [Online]. Available: http://www.theregister.co.uk/2009/02/02/low_cost_rfid_cloner/

[50] ——, "Hacker war drives san francisco cloning rfid passports," [Online; accessed 19-June-2011]. [Online]. Available: http://www.engadget.com/2009/02/02/video-hacker-war-drives-san-francisco-cloning-rfid-passports/

[51] R. Journal, "Rfid - frequently asked questions," [Online; accessed 18-June-2011]. [Online]. Available: http://www.rfidjournal.com/faq/20

[52] J. Bravo, R. Hervás, G. Chavira, S. Nava, and V. Villarreal, "From implicit to touching interaction: Rfid and nfc approaches," in *Human System Interactions, 2008 Conference on*. IEEE, 2008, pp. 743–748.

[53] M. Sallinen, E. Strömmer, and A. Ylisaukko-oja, "Application scenario for nfc: Mobile tool for industrial worker," in *Proceedings of the 2008 Second International Conference on Sensor Technologies and Applications*. Washington, DC, USA: IEEE Computer Society, 2008, pp. 586–591. [Online]. Available: http://portal.acm.org/citation.cfm?id=1446297.1446547

[54] E. Haselsteiner and K. Breitfuß, "Security in near field communication (nfc)," in *Workshop on RFID Security*, 2006.

[55] M. R. Rieback, B. Crispo, and A. S. Tanenbaum, "Is your cat infected with a computer virus?" in *Proceedings of the Fourth Annual IEEE International Conference on Pervasive Computing and Communications*. Washington, DC, USA: IEEE Computer Society, 2006, pp. 169–179. [Online]. Available: http://portal.acm.org/citation.cfm?id=1128015.1128337

[56] C. Mulliner, "Vulnerability analysis and attacks on nfc-enabled mobile phones," in *Availability, Reliability and Security, 2009. ARES '09. International Conference on*, 2009, pp. 695 –700.

[57] C. Leong, K. Ong, K. Tan, and O. Gan, "Near field communication and bluetooth bridge system for mobile commerce," in *Industrial Informatics, 2006 IEEE International Conference on*, 2006, pp. 50 –55.

[58] M. Pasquet, J. Reynaud, and C. Rosenberger, "Secure payment with nfc mobile phone in the smarttouch project," in *Collaborative Technologies and Systems, 2008. CTS 2008. International Symposium on*, May 2008, pp. 121 –126.

[59] J. Gao, L. Prakash, and R. Jagatesan, "Understanding 2d-barcode technology and applications in m-commerce - design and implementation of a 2d barcode processing solution," in *Computer Software and Applications Conference, 2007. COMPSAC 2007. 31st Annual International*, vol. 2, july 2007, pp. 49 –56.

[60] J. Meng and Y. Yang, "Application of mobile 2d barcode in china," in *Wireless Communications, Networking and Mobile Computing, 2008. WiCOM '08. 4th International Conference on*, oct. 2008, pp. 1 –4.

[61] "2d barcode: Data matrix ecc200," [Online; accessed 23-July-2011]. [Online]. Available: http://www.tec-it.com/en/support/knowbase/symbologies/datamatrix/Default.aspx

[62] "About high capacity color barcode technology," [Online; accessed 23-July-2011]. [Online]. Available: http://research.microsoft.com/en-us/projects/hccb/about.aspx

[63] B. Schneier, *Applied cryptography (2nd ed.): protocols, algorithms, and source code in C*. New York, NY, USA: John Wiley & Sons, Inc., 1995.

[64] "An introduction to pki (public key infrastructure)," [Online; accessed 25-June-2011]. [Online]. Available: http://www.articsoft.com/public_key_infrastructure.htm

[65] B. Song, I.-K. Yu, J. Son, and D.-K. Baik, "An effective access control mechanism in home network environment based on spki certificates," in *Information Theory and Information Security (ICITIS), 2010 IEEE International Conference on*, 2010, pp. 592 –595.

[66] T. Saito, K. Umesawa, and H. Okuno, "Privacy enhanced access control by spki," in *Parallel and Distributed Systems: Workshops, Seventh International Conference on, 2000*, Oct. 2000, pp. 301 –306.

[67] "Research firm increases nfc phone forecast on heels of isis shift," [Online; accessed 30-June-2011]. [Online]. Available: http://nfcretailers.com/?p=89

[68] J. Yarow, "Android share surges again, apple gains a bit, rim destroyed," [Online; accessed 30-June-2011]. [Online]. Available: http://www.businessinsider.com/smartphone-market-share-comscore-2011-6

[69] "Android dominance of worldwide smartphone sales goes on, says canalys," [Online; accessed 30-June-2011]. [Online]. Available: http://www.guardian.co.uk/technology/blog/2011/may/04/android-smartphone-worldwide-dominates

[70] "Android activations now total 500,000 a day: Google," [Online; accessed 30-June-2011]. [Online]. Available: http://www.csmonitor.com/Innovation/Horizons/2011/0628/Android-activations-now-total-500-000-a-day-Google

[71] E. Barker, W. Barker, and W. Burr, "Recommendation for key management - part 1 : General," *Nist Special Publication 80057*, no. 1/3, pp. 1–142, 2007. [Online]. Available: http://csrc.nist.gov/publications/nistpubs/800-57/sp800-57-Part1-revised2_Mar08-2007.pdf

[72] R. Woodings, D. Joos, T. Clifton, and C. Knutson, "Rapid heterogeneous ad hoc connection establishment: accelerating bluetooth inquiry using irda," in *Wireless Communications and Networking Conference, 2002. WCNC2002. 2002 IEEE*, vol. 1. IEEE, 2002, pp. 342–349.

[73] G. Chakraborty, K. Naik, D. Chakraborty, N. Shiratori, and D. Wei, "Analysis of the bluetooth device discovery protocol," *Wireless Networks*, vol. 16, no. 2, p. 436, 2010.

[74] D. Scott, R. Sharp, A. Madhavapeddy, and E. Upton, "Using visual tags to bypass bluetooth device discovery," *SIGMOBILE Mob. Comput. Commun. Rev.*, vol. 9, pp. 41–53, January 2005.

[75] A. S. Huang and L. Rudolph, *Bluetooth Essentials for Programmers*. Cambridge: Cambridge University Press, 2007.

[76] A. Varshavsky, A. Scannell, A. LaMarca, and E. De Lara, "Amigo: proximity-based authentication of mobile devices," in *Proceedings of the 9th international conference on Ubiquitous computing*, ser. UbiComp '07. Berlin, Heidelberg: Springer-Verlag, 2007, pp. 253–270. [Online]. Available: http://portal.acm.org/citation.cfm?id=1771592.1771607

[77] M. Laukkanen, "Towards operating identity-based nfc services," in *Pervasive Services, IEEE International Conference on*, 2007, pp. 92 –95.

[78] J.-S. Lee, Y.-W. Su, and C.-C. Shen, "A comparative study of wireless protocols: Bluetooth, uwb, zigbee, and wi-fi," in *Industrial Electronics Society, 2007. IECON 2007. 33rd Annual Conference of the IEEE*, 2007, pp. 46 –51.

[79] C. S. Misal, "Analysis of power consumption of and end device in a zigbee mesh network," Master's thesis, 2007.

[80] T. Kennedy and R. Hunt, "A review of wpan security: attacks and prevention," in *Proceedings of the International Conference on Mobile Technology, Applications, and Systems*, ser. Mobility '08. New York, NY, USA: ACM, 2008, pp. 56:1–56:8. [Online]. Available: http://doi.acm.org/10.1145/1506270.1506342

[81] G. Caronni, "Walking the web of trust," in *Enabling Technologies: Infrastructure for Collaborative Enterprises, 2000. (WET ICE 2000). Proeedings. IEEE 9th International Workshops on*, 2000, pp. 153 –158.

[82] J. Linsky, "Bluetooth and power consumption: issues and answers," 2001.